



TESIS - EE185401

SISTEM PENGHITUNG SKOR TEMBAK OTOMATIS BERBASIS PEMROSESAN CITRA

PARAMA DIPTYA WIDAYAKA
07111650042005

DOSEN PEMBIMBING
Dr. Ir. Hendra Kusuma, M. Eng. Sc.
Muhammad Attamimi, B. Eng, M. Eng, Ph. D

PROGRAM MAGISTER
BIDANG KEAHLIAN TEKNIK ELEKTRONIKA
DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI ELEKTRO
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2019



TESIS - EE185401

SISTEM PENGHITUNG SKOR TEMBAK OTOMATIS BERBASIS PEMROSESAN CITRA

PARAMA DIPTYA WIDAYAKA
07111650042005

DOSEN PEMBIMBING
Dr. Ir. Hendra Kusuma, M. Eng. Sc.
Muhammad Attamimi, B. Eng, M. Eng, Ph. D

PROGRAM MAGISTER
BIDANG KEAHLIAN TEKNIK ELEKTRONIKA
DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI ELEKTRO
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2019

LEMBAR PENGESAHAN

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Teknik (M.T)
di
Institut Teknologi Sepuluh Nopember

oleh:

Parama Diptya Widayaka
NRP. 07111650042005

Tanggal Ujian : 20 Desember 2018
Periode Wisuda : Maret 2019

Disetujui oleh:

1. Dr. Ir. Hendra Kusuma, M. Eng. Sc.
NIP: 196409021989031003

(Pembimbing I)

2. Muhammad Attamimi, B. Eng, M. Eng, Ph. D
NPP: 1985201711039

(Pembimbing II)

3. Dr. Muhammad Rivai, S.T., M.T.
NIP: 196904261994031003

(Penguji)

4. Ronny Mardiyanto, S.T., M.T., Ph. D
NIP: 198101182003121003

(Penguji)

5. Astria Nur Irfansyah, S.T., M.Eng., Ph. D
NIP: 198103252010121002

(Penguji)



Dekan Fakultas Teknologi Elektro

Dr. Tri Arief Sardjono, S.T., M.T.
NIP. 197002121995121001

Halaman ini sengaja dikosongkan

PERNYATAAN KEASLIAN TESIS

Dengan ini saya menyatakan bahwa isi keseluruhan Tesis saya dengan judul **“SISTEM PENGHITUNG SKOR TEMBAK OTOMATIS BERBASIS PEMROSESAN CITRA”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Desember 2018



Parama Diptya Widayaka

NRP. 07111650042005

Halaman ini sengaja dikosongkan

SISTEM PENGHITUNG SKOR TEMBAK OTOMATIS BERBASIS PEMROSESAN CITRA

Nama mahasiswa : Parama Diptya Widayaka
NRP : 07111650042005
Pembimbing : 1. Dr. Ir. Hendra Kusuma, M. Eng. Sc.
2. Muhammad Attamimi, B. Eng, M. Eng, Ph. D

ABSTRAK

Saat ini, masih terdapat lapangan latihan menembak yang menggunakan metode konvensional untuk melakukan penilaian skor dalam kegiatan latihan menembak. Metode konvensional yang digunakan pada kegiatan latihan menembak ini membutuhkan lebih banyak sumber daya manusia dalam melakukan proses penilaian terhadap bekas tembakan. Metode konvensional tersebut juga membutuhkan lebih banyak waktu untuk melakukan penghitungan skor dan. Pada penelitian ini dikembangkan sebuah sistem penghitung skor tembak otomatis berdasarkan pemrosesan citra. Sebuah kamera dipasang pada bagian depan atas terhadap bingkai sasaran tembak yang berfungsi untuk menangkap setiap gambar sasaran tembak pada setiap adanya lubang bekas tembakan baru. Pada penelitian digunakan beberapa algoritma pemrosesan citra seperti transformasi perspektif, pendeteksian lingkaran penilaian, subtraksi citra, serta pengolahan gambar dengan proses morfologi. Pada pemrosesan citra, metode deteksi kontur digunakan untuk membantu dalam melakukan proses transformasi perspektif, pendeteksian lingkaran penilaian dan ekstraksi lubang bekas tembakan. Untuk mendapatkan diameter lingkaran penilaian dan titik pusat lingkaran dilakukan dengan melakukan pembungkakan menggunakan fungsi bounding box pada openCV dari hasil ekstraksi kontur yang terdeteksi pada lingkaran penilaian. Subtraksi citra dilakukan untuk mendeteksi adanya lubang bekas tembakan dengan membandingkan gambar sebelum tembakan dan sesudah adanya tembakan kemudian dilakukan proses deteksi kontur dan pembungkakan dalam mendapatkan titik tengah bekas tembakan untuk kemudian melakukan penilaian. Percobaan dengan menggunakan stiker lingkaran berdiameter 7.62 mm sebagai simulasi bekas lubang tembakan dan percobaan menggunakan 10 set citra dimana setiap set terdiri dari 10 citra dengan bekas tembakan dan pada percobaan terjadi 9 error dari 100 citra dalam pemrosesan penilaian tembakan. Hasil penelitian menunjukkan bahwa tingkat akurasi metode yang telah dikembangkan adalah 91%.

Kata kunci: penghitung skor tembak otomatis, subtraksi citra, transformasi perspektif, deteksi kontur

Halaman ini sengaja dikosongkan

AUTOMATIC SHOOTING SCORING SYSTEM BASED ON IMAGE PROCESSING

By : Parama Diptya Widayaka
Student Identity Number : 07111650042005
Supervisor(s) : 1. Dr. Ir. Hendra Kusuma, M. Eng. Sc.
2. Muh. Attamimi, B. Eng, M. Eng, Ph. D

ABSTRACT

At present, there are shooting venue still use conventional method to make a score in shooting activity. Conventional method which is used in shooting training need more human resources to do scoring process obtained from bullet holes. The method need more time to make a score. This research developed an automatic shooting scoring based on image processing. A camera installed on front top side of target frame to capture image of target sheet every single new bullet hole. This research used some image processing algorithm such as perspective transformation, scoring circle detection, image subtraction, and image processing with morphological processes. In image processing, contour detection method is used to do a perspective transformation process, scoring circle detection and bullet hole extraction. To obtain scoring circle diameter and center position of scoring circle processes by applying framing mechanism using bounding box function in openCV from detected contour extraction of scoring circle. Image subtraction applied to detect bullet holes by comparing image before shot and after shot to apply contour detection and framing mechanism to obtain center position of bullet hole to do scoring process. Circle sticker with 7.62mm diameter as bullet hole simulation is used and this research using 10 set of images which is each set consist of 10 images with bullet holes and the research have 9 errors from 100 total images in scoring process. The experiment result show that accuration method that has been developed is 91%.

Key words: Automatic shooting scoring, image subtraction, perspective transform contour detection

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Alhamdulillah, puji dan syukur penulis panjatkan kepada Allah SWT, karena atas segala petunjuk dan nikmat-Nya lah tesis ini dapat diselesaikan, sholawat dan salam kepada junjungan agung nabi Muhammad SAW serta keluarganya yang suci.

Tesis berjudul “SISTEM PENGHITUNG SKOR TEMBAK OTOMATIS BERBASIS PEMROSESAN CITRA” ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Magister Teknik di Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember.

Penulis menyadari bahwa dalam penyusunan tesis ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, dengan ketulusan dan kerendahan hati penulis menyampaikan terima kasih kepada:

1. Kedua Orang Tua Tercinta penulis, Lilik Yulianto dan Lailatul Qomariyah atas segala dukungan dan doa yang tak pernah putus hingga terselesaikannya tesis penulis.
2. Bapak Dr. Ir. Hendra Kusuma, M. Eng. Sc. dan Bapak Muh. Attamimi, B. Eng, M. Eng, Ph. D selaku Dosen Pembimbing, yang telah banyak memberikan saran dan masukan selama pengerjaan tesis penulis.
3. Bapak Dr. Muhammad Rivai, S.T., M.T., Bapak Ronny Mardiyanto, S.T., M.T., Ph. D, dan Bapak Astria Nur Irfansyah, S.T., M.Eng., Ph. D selaku Dosen Penguji, yang telah memberikan saran dan masukan untuk memperbaiki kekurangan yang terdapat pada tesis penulis.
4. Teman sekelas seperjuangan penulis, Davi, Hadi, dan Aa Reza yang telah banyak membantu, saling mengingatkan dan saling memberikan dukungan selama menyelesaikan tesis.
5. Para sahabat penulis, Fuad, Desna, Sobrun, Rama, Ilham, Ruly, Dani, Dana, dan Latif yang selalu memberi semangat kepada penulis.
6. Teman-teman Sekolah Robot Indonesia Mas Dhadhang, Mas Tri, Mas Ali Zainal, Mas Eric, dan Mas Ali Uroidhi yang juga banyak memberikan bantuan pada penulis.

7. Teman-teman Lab B402, Mas Agung, Juno, Rikho, Wahyu, Anjas, dan Drajat yang selalu menemani dan berbagi cerita bersama penulis.
8. Seluruh Civitas Akademika Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Sepuluh Nopember Surabaya, atas segala bantuan yang telah diberikan.

Semoga Tuhan Yang Maha Pengasih lagi Maha Penyayang membalas semua kebaikan mereka. Pada akhirnya, penulis menyadari bahwa tesis ini masih jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun mengenai tesis yang telah penulis lakukan. Penulis berharap semoga dengan adanya tesis ini dapat menjadi manfaat bagi pengembangan ilmu pengetahuan dan teknologi serta bagi masyarakat demi kemajuan Bangsa dan Negara.

Surabaya, Desember 2018

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN.....	iii
PERNYATAAN KEASLIAN TESIS.....	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan.....	3
1.4 Batasan Masalah	3
1.5 Kontribusi	3
BAB 2 KAJIAN PUSTAKA.....	5
2.1 Kajian Penelitian Terkait.....	5
2.1.1 <i>Evolutionary Low Cost Visual Shooting Practice System (2015)</i>	5
2.1.2 <i>An Automatic Scoring System for Air Pistol Shooting Competition based on Image Recognition of Target Sheets (2015)</i>	6
2.1.3 <i>Implementation of Sensor on the Gun System Using Embedded Camera for Shooting Training (2014)</i>	7
2.1.4 <i>Design of Automatic Target Scoring System of Shooting Game Based On Computer Vision (2009)</i>	8
2.2 Teori Dasar Kegiatan Menembak dan Perangkat Pendukung	9
2.2.1 <i>International Shooting Sport Federation</i>	9
2.2.2 Lembar Sasaran Tembakan.....	10
2.2.3 Perangkat Kamera	11
2.2.4 Mikrokontroler Arduino	12
2.2.5 Modul Sensor Microphone	13
2.2.6 Computer Monitor.....	14

2.3	Teori Dasar Pemrosesan Citra	15
2.3.1	<i>Image Processing</i> atau Pemrosesan Citra.....	15
2.3.2	Jenis Citra.....	16
2.3.3	<i>Image Morphology</i>	19
2.3.4	<i>Perspective Transformation</i> atau transformasi perspektif.....	20
2.3.5	<i>Background Subtraction</i>	20
BAB 3 METODE PENELITIAN		21
3.1	Studi pustaka dan Literatur	22
3.2	Perancangan dan Peletakan <i>Frame</i> Penghitung Skor Otomatis.....	22
3.2.1	Perancangan <i>Frame</i> Penghitung Skor Tembak Otomatis.....	22
3.2.2	Peletakam <i>Frame</i> Penghitung Skor Tembak Otomatis	25
3.3	Perancangan Sistem Penilaian Otomatis	26
3.3.1	Proses <i>Image Transformation</i> pada Sasaran Tembak	28
3.3.2	Proses <i>Ring Detection</i> pada Sasaran Tembak	35
3.3.3	Proses <i>Bullet Hole Detection</i> pada Sasaran Tembak.....	38
3.3.4	<i>Scoring Mechanism</i>	42
BAB 4 HASIL DAN PEMBAHASAN		43
4.1	Pengujian penilaian simulasi tembakan menggunakan stiker.....	43
4.1.1	Pengujian Simulasi Tembakan set Ke-1	43
4.1.2	Pengujian Simulasi Tembakan set Ke-2	43
4.1.3	Pengujian Simulasi Tembakan Set Ke-3	44
4.1.4	Pengujian Simulasi Tembakan set Ke-4.....	44
4.1.5	Pengujian Simulasi Tembakan Set Ke-5	44
4.1.6	Pengujian Simulasi Tembakan Set Ke-6	45
4.1.7	Pengujian Simulasi Tembakan Set Ke-7	45
4.1.8	Pengujian Simulasi TembakanSset Ke-8.....	45
4.1.9	Pengujian Simulasi Tembakan Set Ke-9	46
4.1.10	Pengujian Simulasi Tembakan Set Ke-10	46
4.1.11	Analisa Data Percobaan	47
BAB 5 PENUTUP		49
5.1	Kesimpulan	49
5.2	Saran	49

DAFTAR PUSTAKA	51
LAMPIRAN	53
BIOGRAFI PENULIS	85

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2. 1 Sasaran tembak terstandarisasi PERBAKIN	10
Gambar 2. 2 Spesifikasi Sasaran Tembak Kategori Pistol.....	10
Gambar 2. 3 Sasaran Tembak Kategori Rifle.....	11
Gambar 2. 4 Susunan Kamera	11
Gambar 2. 5 Arduino Mega 2560 R3.....	12
Gambar 2. 6 Modul mikrofon dfrobot DFR0034	13
Gambar 2. 7 Computer Monitor	14
Gambar 2. 8 Citra berwarna	16
Gambar 2. 9 Citra Grayscale	17
Gambar 2. 10 Citra Biner	18
Gambar 2. 11 Proses operasi erosi.....	19
Gambar 2. 12 Proses operasi dilasi	19
Gambar 2. 13 Gambar plat yang terdistorsi	20
Gambar 3. 1 Diagram blok tahapan penelitian	21
Gambar 3. 2 Desain frame skor tembak otomatis	23
Gambar 3. 3 Frame skor tembak otomatis	23
Gambar 3. 4 Posisi peletakan kamera pada frame	24
Gambar 3. 5 Peletakan frame penghitung skor otomatis terhadap penembak	26
Gambar 3. 6 Diagram blok sistem penghitung skor tembak otomatis	27
Gambar 3. 7 Hasil pengambilan gambar sasaran tembak	28
Gambar 3. 8 Gambar sasaran tembak setelah dilakukan proses rotasi	29
Gambar 3. 9 Blok diagram proses transformasi perspektif.....	29
Gambar 3. 10 Citra sasaran tembak setelah dilakukan proses grayscale	30
Gambar 3. 11 Hasil thresholding pada citra sasaran tembak	31
Gambar 3. 12 Hasil pendeteksian kontur lembar sasaran tembak	32
Gambar 3. 13 Hasil proses cropping terhadap sasaran tembak.....	33
Gambar 3. 14 Deteksi kontur dan pembingkai pada sasaran tembak	34
Gambar 3. 15 Hasil transformasi citra sasaran tembak.....	34
Gambar 3. 16 Blok diagram pendeteksian lingkaran penilaian.....	35
Gambar 3. 17 Hasil deteksi kontur citra sasaran tembak	35
Gambar 3. 18 Hasil bounding box terhadap kontur lingkaran	36
Gambar 3. 19 Lingkaran penilaian hasil ekstraksi bounding box	37
Gambar 3. 20 Blok diagram pendeteksian bekas tembakan.....	38
Gambar 3. 21 Perbedaan citra sasaran tembak	39
Gambar 3. 22 Hasil subtraksi citra sasaran tembak	40
Gambar 3. 23 Hasil proses filter morfologi bekas tembakan	41
Gambar 3. 24 Bekas tembak terdeteksi dengan diberi bingkai merah.....	41
Gambar 3. 25 Representasi posisi bekas tembakan	42

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 2.1 Kategori Olimpiade Menembak	9
Tabel 4.1 Hasil Percobaan Simulasi Set Ke-1	43
Tabel 4.2 Hasil Percobaan Simulasi Set Ke-2	43
Tabel 4.3 Hasil percobaan simulasi set Ke-3	44
Tabel 4. 4 Hasil percobaan simulasi set Ke-4	44
Tabel 4.5 Hasil percobaan simulasi set Ke-5	44
Tabel 4.6 Hasil percobaan simulasi set Ke-6	45
Tabel 4.7 Hasil percobaan simulasi set Ke-7	45
Tabel 4.8 Hasil percobaan simulasi set Ke-8	45
Tabel 4.9 Hasil percobaan simulasi set Ke-9	46
Tabel 4.10 Hasil percobaan simulasi set Ke-10	46

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Sebelum digunakannya senjata api pada tahun 1300, kegiatan menembak pertama kali dilakukan sebagai suatu ajang tes keterampilan yang dimulai dengan panahan. Penggunaan senjata api yang pertama kali diperuntukkan dalam perang di medan pertempuran dan seiring berjalannya waktu kemudian senjata api digunakan dalam kegiatan olahraga menembak atau berburu. Pada tahun 1477, sebuah senjata api mulai digunakan dalam kegiatan perlombaan menembak yang dilaksanakan Eichstaat, Bassvaria, dimana dalam perlombaan ditentukan dengan jarak menembak sejauh 200 meter dengan menggunakan *matchclock* untuk setiap atlet [1]. Pada saat ini, kegiatan tembak menembak tidak hanya dikhususkan bagi para angkatan bersenjata atau tentara tetapi juga menjadi bagian dari sebuah perlombaan yang bertaraf nasional maupun internasional dengan beberapa kategori dan selain yang disebutkan disisi lain para pemburu dan penghobi kegiatan menembak membutuhkan sarana yang mendukung dalam latihan menembak untuk sekedar bersenang-senang ataupun untuk latihan pertahanan diri [2]. Dengan semakin berkembangnya zaman, perkembangan pada kegiatan menembak semakin maju dan lebih bervariasi pada sisi senjata api yang digunakan dan yang utama pada sisi sistem penilaian kegiatan menembak. Sejalan dengan perkembangan teknologi, mekanisme penilaian untuk kegiatan menembak semakin berkembang ditandai dengan digunakannya sistem otomatis pada mekanisme penilaian tembakan. Sistem penilaian tembakan otomatis dibedakan menjadi beberapa kategori yaitu sistem penilaian berbasis akustik, sistem penilaian berbasis visual, dan perpaduan antara sistem akustik dengan sistem visual [3]. Salah satu sistem penilaian menembak otomatis yang sedang berkembang adalah sistem penilaian dengan menggunakan teknologi berbasis visual. Sistem penilaian menembak otomatis yang berbasis visual menggunakan perangkat kamera sebagai sensor untuk mendapatkan informasi mengenai hasil tembakan yang dilakukan oleh atlet pada saat melakukan latihan menembak atau dapat juga diaplikasikan pada kegiatan perlombaan

menembak. Pada beberapa keadaan, informasi mengenai hasil latihan yang diperoleh oleh atlet tidak dapat dilihat secara langsung dikarenakan kurangnya sistem yang memfasilitasi monitoring secara langsung [4] dan disisi lain, sistem penilaian dengan metode konvensional yang dioperasikan atau dibantu dengan tenaga manusia kurang efisien, keamanan saat melakukan penilaian, dan yang terakhir merupakan konsumsi waktu yang diperlukan dalam proses penilaian [5]. Banyak peneliti yang melakukan penelitian dibidang visual terutama pemrosesan citra seperti contohnya robot sepak bola yang menggunakan camera untuk mendeteksi posisi bola dan letak gawang untuk melakukan tendangan. Pada penelitian kali ini, sebuah sistem yang berbasis visual atau pemrosesan citra digunakan untuk mendeteksi lingkaran penilaian yang terdapat pada lembar sasaran tembak dan menentukan posisi dari jejak tembakan yang terdapat pada lingkaran penilaian terhadap lembar sasaran tembak. Sistem penilaian tembakan otomatis merupakan suatu sistem yang dibutuhkan pada saat ini untuk memfasilitasi kegiatan menembak pada kegiatan latihan atau perlombaan seorang atlet menembak, pemburu ataupun para penghobi kegiatan menembak. Latar belakang pertama dari dilakukannya penelitian ini adalah sistem kalkulasi dan penilaian pada kegiatan menembak adalah adanya beberapa lapangan latihan menembak yang masih menggunakan cara sederhana yang dilakukan dengan manual dimana cara penilaian tersebut memerlukan tambahan waktu untuk menentukan dan mengumpulkan data tembakan untuk melakukan penilaian selain hal itu, cara penilaian yang sederhana tersebut memerlukan sumber daya tambahan untuk melakukan proses penilaian [6]. Latar belakang kedua, biaya untuk sebuah alat penghitung skor tembakan otomatis tidaklah murah sehingga hanya beberapa tempat latihan saja yang menggunakan sistem skor otomatis untuk keperluan latihan menembak maupun kegiatan perlombaan seperti contohnya hanya ada beberapa daerah yang menggunakan sistem otomatis tersebut yaitu DKI Jakarta dan Sumatera Selatan [7]. Diharapkan dengan menggunakan sistem penghitung skor tembakan otomatis dapat memfasilitasi kegiatan latihan menembak bagi para atlet sehingga dapat meningkatkan performa dan juga dapat memberikan timbal balik yang baik kepada atlet menembak [8].

1.2 Rumusan Masalah

1. Bagaimana menentukan metode yang digunakan untuk mendeteksi sasaran atau target tembak.
2. Bagaimana menentukan metode yang digunakan untuk mendeteksi jejak atau bekas peluru yang mengenai sasaran atau target tembak.
3. Bagaimana proses perhitungan atau *scoring* dengan memanfaatkan pembacaan bekas tembakan pada sasaran tembak.

1.3 Tujuan

1. Mendapatkan metode untuk mengenali citra lubang bekas tembakan yang dihasilkan oleh senapan yang ada pada sasaran tembak.
2. Mendapatkan metode untuk memberi penilaian ketepatan menembak berdasarkan posisi citra lubang bekas tembakan.
3. Dapat menampilkan skor atau nilai tembakan secara langsung / *realtime*.

1.4 Batasan Masalah

Dalam pengajuan penelitian, terdapat beberapa batasan masalah yaitu sebagai berikut:

1. Percobaan dilakukan di ruangan dengan intensitas cahaya yang relatif sama.
2. Sasaran tembak bersifat diam ditempat atau *fix*.
3. Sifat tembakan berupa *single shot*.
4. Jumlah tembakan pada sasaran maksimum adalah 10 jejak peluru.
5. Citra lubang tembakan disimulasikan dengan menggunakan stiker lingkaran dengan ukuran diameter 7.62mm.

1.5 Kontribusi

Kontribusi yang diharapkan dari penelitian ini adalah mendapatkan algoritma ketepatan dan kecepatan dalam menentukan nilai tembakan dan juga terciptanya sebuah sistem penghitung skor otomatis yang nantinya dapat diaplikasikan dalam kegiatan tembak-menembak berupa latihan maupun suatu perlombaan dalam mengoptimalkan sistem penilaian atau *scoring*.

Halaman ini sengaja dikosongkan

BAB 2

KAJIAN PUSTAKA

2.1 Kajian Penelitian Terkait

Pada bagian ini akan ditunjukkan penelitian – penelitian terkait tentang sistem penghitung skor tembak otomatis yang pernah dilakukan sebelumnya, sehingga dari penelitian yang telah ada dengan berbagai metode akan menjadi sebuah studi literature yang akan digunakan sebagai acuan dan nantinya bisa mengembangkan penelitian yang akan dilakukan kali ini. Dibawah ini akan ditunjukkan mengenai penelitian-penelitian yang telah dilakukan.

2.1.1 *Evolutionary Low Cost Visual Shooting Practice System (2015)[2]*

Pada tahun 2015, Khaled S. Hatamleh dkk dari *Mechanical Eng. Department Jordan University of Science and Technology*, telah dilakukan sebuah penelitian yang berkaitan tentang sistem latihan menembak yang berbasis visual. Pada penelitian tersebut, pistol yang digunakan untuk kegiatan latihan menembak bukan merupakan senjata yang dapat menembakkan peluru tetapi pada senjata tersebut telah ditanamkan sebuah laser yang kemudian dapat menghasilkan jejak tembakan pada sasaran tembak berupa cahaya laser merah dan telah diberikan sebuah kamera CCD pada sasaran tembak sebagai sensor untuk mendeteksi posisi cahaya merah yang dihasilkan oleh laser tembakan pada sasaran tembak. Perbedaan pada penelitian yang telah dilakukan oleh penulis terhadap penelitian diatas adalah penggunaan senjata yang dilakukan pada penelitian penulis dimana pada penelitian penulis menggunakan senjata asli dengan menggunakan peluru berdiameter 7.62mm sehingga bekas tembakan yang tercetak pada sasaran tembak merupakan sebuah lubang atau sobekan yang dihasilkan oleh peluru dan kemudian dilakukan sebuah proses *Image Processing* pada lembaran sasaran tembak dan jejak tembakan untuk kemudian dilakukan mekanisme penilaian berdasarkan posisi bekas tembakan terhadap lingkaran yang ada pada lembaran sasaran tembak.

2.1.2 *An Automatic Scoring System for Air Pistol Shooting Competition based on Image Recognition of Target Sheets (2015)[6]*

Pada tahun 2015, dilakukan sebuah penelitian tentang pembacaan nilai atau skor tembakan yang tertera pada lembar sasaran tembak dilakukan oleh Yuan-Chi Lin dan tim dari *Department of Electronic Engineering Chung-Yuan University, Taiwan*. Pada penelitian yang telah dilakukan, proses penilaian terhadap bekas tembakan dilakukan dengan cara men-scan lembar sasaran tembak yang di dalamnya terdapat bekas tembakan pada setiap lingkaran skor. Sistem penilaian pada penelitian tersebut dilakukan dengan melakukan 2 langkah pengenalan lembar sasaran tembak yaitu, *Automatic Score Reading* dan *Number Recognition*. Pada proses *Automatic Score Reading* dilakukan sebuah proses pendeteksian nomor seri pada lembar sasaran tembak dengan mendeteksi ROI yang berupa kemungkinan posisi keberadaan nomor seri pada sasaran tembak. Pada proses *automatic score reading*, telah diberikan kertas berwarna biru di bawah lembar sasaran tembak dimana berfungsi untuk memudahkan pendeteksian bekas tembakan yang tertera pada lembar sasaran tembak. Pada proses *automatic score reading*, dilakukan sebuah proses pendeteksian lubang tembakan pada lembar sasaran tembak dengan mendeteksi warna biru terhadap bekas tembakan untuk kemudian dapat ditentukan titik tengah dari bekas tembakan tersebut dan dijadikan data titik tengah berupa koordinat pada sumbu x dan sumbu y terhadap bekas tembakan yang telah dilakukan oleh proses *bullet hole detection* yang nantinya akan digunakan untuk proses penilaian. Setelah dilakukan proses *bullet hole detection*, kemudian dilanjutkan untuk melakukan proses *score calculation* dengan menggunakan rumus *euclidean distance* untuk mendeteksi jarak antara titik tengah bekas tembakan dan titik tengah lingkaran pada lembar sasaran tembak. Pada penelitian ini proses penilaian yang dilakukan adalah dengan men-scan lembar target sehingga pada sistem ini tidak bersifat *realtime* karena penilaian dilakukan setelah selesainya sesi menembak dan perbedaan dengan penelitian yang akan dilakukan penulis adalah sistem penilaian dan pemrosesan citra akan dilakukan secara *realtime* saat sesi menembak.

2.1.3 Implementation of Sensor on the Gun System Using Embedded Camera for Shooting Training (2014)[9]

Pada tahun 2014, Aryunto Sutedjo dan tim dari Institut Teknologi Nasional, Malang, Indonesia melakukan sebuah penelitian deteksi tembakan dimana pada pistol ditanamkan sebuah laser untuk menembakkan sinar laser pada sasaran tembak dan sebuah CMUCAM untuk mendeteksi hasil tembakan berupa sinar yang dihasilkan oleh laser. Pada penelitian tersebut, terdapat empat buah titik di setiap sisi pojok pada target yang menjadi titik acuan untuk melakukan proses transformasi citra untuk memperbaiki citra target yang terdistorsi. Dalam penelitian ini, lembar target yang digunakan tidak spesifik dan tidak mengacu pada lembar sasaran yang biasa digunakan dalam perlombaan menembak karena dalam penelitian tersebut menitik beratkan pada proses pengoreksian citra target yang terdistorsi sehingga citra yang awalnya terdistorsi akan di transformasi menjadi citra yang tegak lurus seperti pengambilan citra yang tegak lurus dengan kamera. Perbedaan pada penelitian ini dengan yang akan dilakukan penulis adalah, pada penelitian yang dilakukan oleh Aryunto Sutedjo dan tim menggunakan empat buah penanda/*marker* berwarna merah pada setiap sudut atau posisi pojok pada lembar sasaran tembak untuk kemudian menjadi acuan dalam proses transformasi citra sasaran tembak, sedangkan dalam penelitian yang telah dilakukan oleh penulis, dalam proses transformasi citra dilakukan dengan cara mendeteksi kontur lembar sasaran tembak yang berbentuk kotak untuk kemudian dilakukan proses *bounding box* yang terdapat pada fungsi di openCV yang nantinya akan dijadikan acuan untuk mengkoreksi citra sasaran tembak yang terdistorsi menjadi citra yang tegak lurus sehingga dalam proses pendeteksian lingkaran penilaian pada lembar sasaran tembak dapat dilakukan dengan mudah sehingga dari lingkaran yang ada pada sasaran tembak dapat ditentukan besar diameter pada setiap lingkaran yang nantinya akan digunakan sebagai acuan penilaian bekas tembakan terhadap setiap lingkaran penilaian yang ada pada sasaran tembak.

2.1.4 *Design of Automatic Target Scoring System of Shooting Game Based On Computer Vision (2009)[4]*

Pada tahun 2009, telah dilakukan penelitian berkaitan tentang sistem penilaian sasaran otomatis oleh Xinnan Fan dan tim dari Computer and Information Institute, Hohai University, China. Pada penelitian tersebut telah dibuat sebuah sistem penilaian otomatis dengan berbasis pemrosesan citra. Sistem otomatis tersebut diberikan sebuah kamera dibagian depan bawah frame dan dibagian belakang diberikan cahaya untuk memudahkan dalam pendeteksian jejak tembakan pada sasaran tembak. Dalam penelitian yang dilakukan oleh Xinnan dan tim terdapat beberapa proses untuk melakukan penilaian yaitu pendeteksian lingkaran target dan pendeteksian posisi bekas tembakan pada lembar sasaran target. Pendeteksian lingkaran target yang telah dilakukan adalah dengan cara mendeteksi lingkaran target yang berwarna hitam yaitu lingkaran paling dalam pada sasaran tembak untuk didapatkan garis tepi lingkaran tersebut kemudian dari lingkaran tersebut dapat diekstrak menjadi beberapa lingkaran yang lain dengan mendeteksi kontur pada lingkaran. Pada penelitian ini terdapat beberapa perbedaan dengan penelitian mengenai sistem penghitung skor otomatis yang telah dilakukan penulis dimana pada penelitian yang dilakukan oleh Xinnan dan tim tidak dijelaskan ada atau tidaknya proses transformasi melihat posisi kamera yang ada pada frame berada pada bagian bawah frame sehingga gambar yang ditangkap seharusnya terdistorsi dan selain hal itu, pendeteksian yang dilakukan oleh penulis selain menggunakan deteksi kontur dilanjutkan dengan menggunakan fungsi bounding box terhadap kontur setiap lingkaran penilaian untuk kemudian dapat diakses lebar dan tinggi dari setiap lingkaran penilaian berdasarkan sumbu x dan sumbu y sehingga dapat ditentukan titik tengah dari setiap lingkaran penilaian dan juga nilai diameter setiap lingkaran untuk dijadikan data acuan penilaian terhadap bekas tembakan yang terdeteksi pada lembar sasaran tembakan.

2.2 Teori Dasar Kegiatan Menembak dan Perangkat Pendukung

2.2.1 *International Shooting Sport Federation*

Selama abad ke-19, bentuk organisasi sebelumnya seperti klub menembak berkembang menjadi federasi menembak nasional. *Société Suisse des Carabiniers* didirikan pada tahun 1824 di Swiss. Ratu Victoria meresmikan *British National Rifle Association* pada tahun 1859. Duke Ernst II dari Saxony-Coburg dan Gotha mendirikan *Deutscher Schützenbund* pada tahun 1861, dan para pemimpin militer mendirikan *National Rifle Association of America* (AS) pada tahun 1871. Federasi *Francaise des Societies de Tir* secara resmi didirikan pada tahun 1884, namun federasi nasional Prancis sebelumnya telah mengawalinya. Olahraga menembak menjadi bagian dari olimpiade modern sejak pertama kali diperkenalkan. Pada olimpiade pertama, yang diselenggarakan di Athena pada tahun 1896, 39 penembak dari tujuh negara berkompetisi dalam tiga kategori pistol dan dua senapan berdaya tinggi. Menembak juga termasuk dalam program kedua olimpiade dimana diikuti 139 orang penembak dari 13 negara yang berkompetisi dalam tiga kategori pistol, lima rifle, dua target lilin dan satu sasaran bergerak. dengan sejalannya waktu, hingga saat ini dalam olimpiade menembak terdapat 390 atlet menembak dari 97 negara yang telah dikualifikasi untuk berkompetisi dalam 15 rifle, pistol dan shotgun pada kegiatan Rio 2016 *Olympic Games* yang dimulai dari tanggal 6 hingga 14 agustus 2016 [10]. Pada tabel 2.1 merupakan kategori dalam olimpiade menembak.

Tabel 2.1 Kategori Olimpiade Menembak

Disciplines	Men's Events	Women's Events	Total Events
Rifle	50m Rifle 3 Position, 50m Rifle Prone, 10m Air Rifle	50m Rifle 3 Positions, 10m Air Rifle	5
Pistol	50m Pistol, 50m Rapid Fire Pistol, 10m Air Pistol	25m Pistol, 10m Air Pistol	5
Shotgun	Trap, Double Trap, Skeet	Trap, Skeet	5
Total	9	6	15




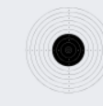
2.2.2 Lembar Sasaran Tembakan

Target tembak adalah sebuah objek dengan bentuk yang beragam dan digunakan untuk pistol, rifle, shotgun dan olahraga menembak yang lain, seperti contohnya darts, sasaran panahan, tembak busur dan olahraga non-senapan yang lain. Pada bagian tengah disebut sebagai *bullseye*. Sasaran tembak dapat dibuat secara langsung dengan menggunakan kertas, karet “*self healing*” atau besi. Selain itu, juga terdapat sasaran tembak yang bersifat elektronik yang dapat memberikan timbal balik yang presisi kepada penembak mengenai letak peluru [11].



Gambar 2.1 Sasaran tembak terstandarisasi PERBAKIN

Pada Gambar 2.2 dibawah ini merupakan spesifikasi dari target atau sasaran tembak kategori pistol yang banyak digunakan dalam perlombaan internasional maupun nasional.

								
	25 m Rapid Fire Pistol		25 m Precision Pistol		50 Meter Pistol		10 Meter Air Pistol	
	Subtension	Angular size	Subtension	Angular size	Subtension	Angular size	Subtension	Angular size
Inner Ten	50 mm	2 mil	25 mm	1 mil	25 mm	0.5 mil	5 mm	0.5 mil
10 Ring	100 mm	4 mil	50 mm	2 mil	50 mm	1 mil	11.5 mm	1.15 mil
Subsequent Ring Increase	80 mm	3.2 mil	50 mm	2 mil	50 mm	1 mil	16 mm	1.6 mil

Gambar 2. 2 Spesifikasi Sasaran Tembak Kategori Pistol

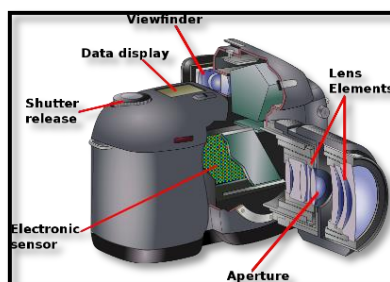
Dan untuk gambar 2.3 dibawah ini merupakan spesifikasi sasaran tembak untuk senjata kategori *rifle*.

	300 m Rifle		50 m Rifle		10 m Air Rifle	
	Subtension	Angular size	Subtension	Angular size	Subtension	Angular size
Inner Ten	50 mm	1/6 mil = 0.167 mil	5 mm	0.1 mil	(Determined by scoring gauge)	-
10 Ring	100 mm	1/3 mil = 0.33 mil	10.4 mm	0.208 mil	0.5 mm	0.05 mil
Subsequent Ring Increase	100 mm	1/3 mil = 0.33 mil	16 mm	0.32 mil	5 mm	0.5 mil

Gambar 2. 3 Sasaran Tembak Kategori Rifle

2.2.3 Perangkat Kamera

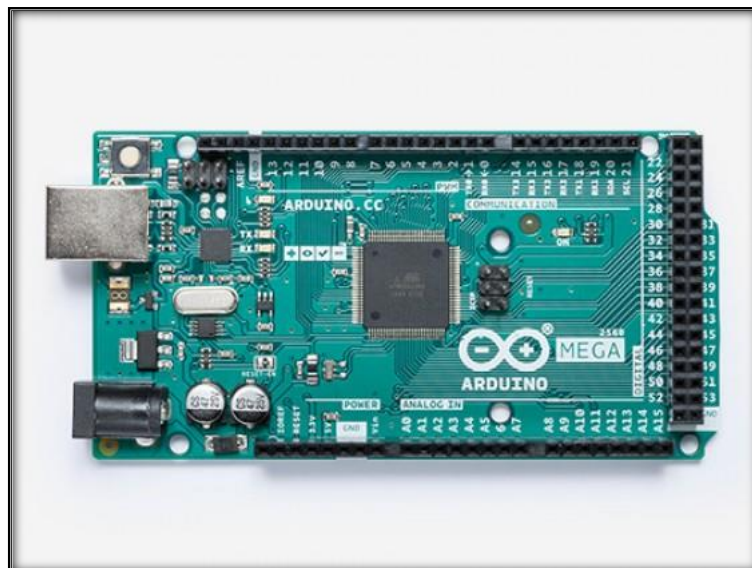
Kamera adalah perangkat optik untuk merekam maupun menangkap gambar, dimana dapat disimpan langsung, dipindahkan ke lokasi yang lain, atau juga keduanya. Sebuah gambar dapat berupa sebuah foto atau juga berupa urutan gambar yang merupakan video atau film. Kamera adalah sebuah alat penginderaan jauh dimana dapat melihat adanya subyek tanpa adanya kontak langsung. Kata kamera berasal dari kamera obscura, yang berarti “ruang gelap” dan merupakan nama latin dari sebuah perangkat untuk memproyeksikan sebuah citra pada permukaan yang rata. Kamera fotografi modern berevolusi dari kamera obscura. Fungsi dari kamera sama dengan fungsi dari mata manusia. Foto permanen pertama yang dihasilkan oleh kamera dibuat pada tahun 1826 oleh Joseph Nicéphore Niépce. Sebuah kamera dapat bekerja dengan cahaya spectrum yang terlihat atau dengan bagian spektrum elektromagnetik lainnya. Semua kamera menggunakan rancangan dasar yang sama dimana cahaya memasuki kotak tertutup melalui lensa cembung atau cekung dan gambar akan ditangkap pada media yang peka terhadap cahaya seperti contohnya logam transisi halide [12].



Gambar 2. 4 Susunan Kamera

2.2.4 Mikrokontroler Arduino

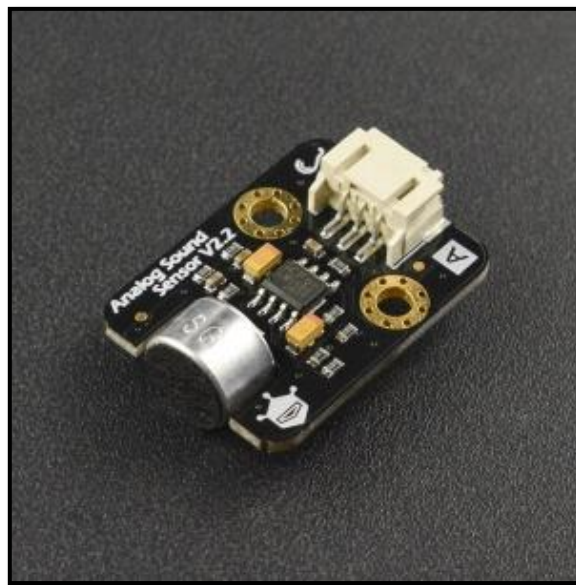
Mikrokontroler merupakan sebuah sistem benam yang berfungsi untuk mengolah data-data digital mapunu data analog yang didalamnya merupakan gabungan dari rangkaian elektronik yang dikemas dalam suatu *chip* (IC). Pada umumnya, mikrokontroler terdiri dari beberapa bagian seperti alamat (*address*), data, pengendali, memori (RAM atau ROM), dan bagian pada mikrokontroler yang berfungsi sebagai input dan output. Mikrokontroler dirancang untuk aplikasi embedded, sangat jauh berbeda dengan mikroprosesor yang digunakan untuk untuk komputer pribadi atau aplikasi umum lainnya. Mikrokontroler biasa digunakan dalam produk yang menggunakan sistem otomasi seperti sistem kontrol mesin mobil, perangkat medis, remot kontrol, alat listrik dan lainnya. Pada penelitian yang dilakukan oleh penulis, mikrokontroler yang digunakan untuk mendukung sistem benam dari sistem penghitung skor tembak otomatis adalah arduino mega 2560 R3 tertera pada gambar 2.5 yang disambungkan dengan modul sensor microphone untuk mendeteksi adanya suara tembakan untuk dijadikan pemicu kamera dalam pengambilan gambar sasaran tembak kemudian dilakukan sebuah pemrosesan citra untuk melakukan penilaian terhadap bekas tembakan yang tercetak pada lembar sasaran tembak.



Gambar 2. 5 Arduino Mega 2560 R3

2.2.5 Modul Sensor Microphone

Pada penelitian yang dilakukan oleh penulis membutuhkan sebuah perangkat tambahan untuk mendukung kinerja dari sistem penghitung skor otomatis yaitu sebuah modul sensor microphone yang berfungsi untuk mendeteksi adanya suara yang dihasilkan oleh senjata pada saat melakukan latihan menembak. Modul mikrofon yang digunakan untuk mendeteksi adanya tembakan pada sistem sistem penghitung skor tembak otomatis menggunakan modul mikrofon yang merupakan produk dari dfrobot yang dapat dilihat pada gambar 2.6.



Gambar 2. 6 Modul mikrofon dfrobot DFR0034

Modul mikrofon tersebut bekerja pada tegangan 3.3 volt hingga 5 volt dalam pendeteksian intensitas suara. Pada modul mikrofon tersebut terdapat 3 pin untuk sambungan vcc, ground dan data yang nantinya akan disambungkan ke arduino untuk memproses data yang dihasilkan oleh modul mikrofon tersebut. Data yang dihasilkan oleh modul mikrofon tersebut berupa data analog yang memiliki rentang nilai 0 hingga 255 dan pada penelitian yang dilakukan oleh penulis, modul sensor mikrofon digunakan untuk mendeteksi suara tembakan dengan mengamati nilai keluaran dari modul mikrofon yang kemudian nilai tersebut diproses untuk menentukan besar nilai keluaran yang dihasilkan oleh suara tembakan untuk dijadikan pemicu bagi kamera dalam proses penangkapan gambar sasaran tembak.

2.2.6 Computer Monitor

Computer monitor merupakan perangkat keluaran atau *output* dimana berfungsi untuk menampilkan informasi dalam bentuk gambar. Sebuah perangkat monitor biasanya berisi perangkat tampilan, rangkaian, *casing* atau pelindung, dan sumber tenaga. Perangkat tampilan pada *monitor* modern biasanya berupa *display* kristal cair yang terbuat dari lembaran tipis transistor (*thin film transistor liquid crystal display*) atau biasa disingkat menjadi TFT-LCD dengan LED sebagai pencahayaan yang saat ini telah digantikan oleh CCFL (*Cold cathode fluorescent lamp*). *Monitor* terdahulu masih menggunakan CRT (*Cathode ray tube*). *Monitor* dihubungkan melalui VGA, *Digital Visual Interface* (DVI), HDMI, *DisplayPort*, *Thunderbolt*, *Low-voltage differential signalling* (LVDS) atau alat penghubung yang lain. Pada gambar 2.7 merupakan contoh dari *monitor* yang tersedia di pasaran.



Gambar 2. 7 Computer Monitor

2.3 Teori Dasar Pemrosesan Citra

2.3.1 *Image Processing* atau Pemrosesan Citra

Definisi dasar dari suatu citra atau gambar diartikan sebagai suatu fungsi dari dua buah variabel seperti contoh $a(x,y)$ dimana a merupakan amplitudo (contohnya kecerahan) citra pada koordinat (x,y) . Menurut Ian T. Young dkk, citra digital $a[m,n]$ merupakan citra dalam ruang diskrit dua dimensi yang berasal dari citra analog $a(x,y)$ di ruang kontinyu dua dimensi melalui proses sampling yaitu yang biasa disebut sebagai proses digitalisasi. Sedangkan menurut Maria, citra digital adalah citra $f(x,y)$ yang telah didiskritkan pada koordinat spasial dan kecerahan. Citra digital direpresentasikan oleh *array* dua dimensi atau sekumpulan *array* dua dimensi dimana setiap *array* merepresentasikan satu kanal warna. Nilai kecerahan yang didigitalkan dinamakan dengan nilai tingkat keabuan atau *grayscale* [13]. Input dari program pengolahan citra adalah citra dan outputnya juga berupa citra. Pengolahan citra dikembangkan bertujuan untuk:

1. Untuk memperbaiki tampilan citra (*image enhancement*).
2. Untuk mengurangi ukuran *file* citra dengan tetap mempertahankan kualitas citra (*image compression*).
3. Untuk memulihkan citra ke kondisi semula (*image restoration*).
4. Untuk menyoroti ciri tertentu dari citra agar lebih mudah untuk di analisis.

Dewasa ini pengolahan citra digital digunakan dalam berbagai bidang untuk mempermudah manusia dalam melakukan analisis dan pekerjaan. Dibawah ini merupakan bentuk aplikasi pengolahan citra digital yang sering digunakan.

1. Bidang Militer
2. Bidang Industri
3. Bidang Medis
4. Bidang Transportasi
5. Bidang Hukum dan Keamanan
6. Pemetaan
7. Robotika
8. Fotografi, dan lain-lain.

2.3.2 Jenis Citra

Pada pengolahan citra terdapat beberapa jenis citra yang akan dibahas pada bagian ini diantaranya adalah citra berwarna, citra *grayscale* / derajat keabuan, dan citra biner

1. Citra Berwarna

Citra berwarna adalah citra yang memiliki 3 buah kanal warna di dalamnya. Pada umumnya jenis citra ini terbentuk dari komponen merah, hijau dan biru atau banyak yang menyebutnya dengan citra RGB (*Red, Green, Blue*). RGB adalah standar yang digunakan untuk menampilkan citra berwarna pada layar televisi maupun layar komputer. Namun terdapat juga citra berwarna yang menggunakan ruang warna yang berbeda, seperti CMYK, HSV, YCbCr dan Lab[13]. Pada gambar 2.13 di bawah ini merupakan contoh dari citra berwarna.



Gambar 2. 8 Citra berwarna

Citra berwarna atau *true color image* sering juga disebut dengan 24-bit *color image* karena untuk setiap nilai pikselnya memerlukan penyimpanan sebesar 24-bit. Masing-masing piksel dalam setiap kanal memiliki kemungkinan nilai sebanyak 256 kemungkinan, yakni dengan nilai diantara 0-255. Karena citra berwarna memiliki 3 buah kanal, maka untuk satu piksel pada citra berwarna memerlukan $3 \times 8 \text{ bit} = 24 \text{ bit}$, maka dari itu citra berwarna disebut sebagai 24-bit *color image*. Dengan kombinasi warna yang ada, maka citra berwarna memiliki kemungkinan sebanyak $256 \times 256 \times 256 = 2^{24} = 16.777.216$ variasi warna.

2. Citra *Grayscale*

Citra *grayscale* adalah citra yang hanya memiliki 1 buah kanal sehingga yang ditampilkan hanyalah nilai intensitas atau dikenal juga dengan istilah derajat keabuan[13]. Karena citra jenis ini hanya memiliki 1 kanal saja, maka ini disebut juga sebagai 8-bit *image* karena untuk setiap nilai pikselnya memerlukan penyimpanan sebesar 8-bit. Foto hitam putih maupun gambar yang ditampilkan oleh televisi hitam putih sebenarnya menggunakan citra *grayscale*, bukan dalam warna hitam dan warna putih. Pada gambar 2.14 dibawah ini merupakan contoh dari gambar citra *grayscale*.



Gambar 2. 9 Citra Grayscale

Secara teori ada beberapa cara dalam mengonversi citra berwarna RGB ke dalam citra *grayscale*. Cara yang paling mudah adalah dengan merata-ratakan semua nilai piksel RGB sesuai dengan persamaan (1) dibawah ini.

$$y = \frac{1}{3}(R + G + B) \quad (2.1)$$

Namun hasil konversi yang diberikan dengan menggunakan persamaan (2.1) tidak terlalu bagus. Untuk mendapatkan hasil konversi yang lebih baik, persamaan yang dapat digunakan seperti dibawah ini.

$$y = 0.299R + 0.587G + 0.144B \quad (2.2)$$

3. Citra Biner

Citra biner atau juga dapat disebut sebagai citra hitam putih (*black and white image*) adalah citra yang hanya memiliki 2 kemungkinan nilai untuk setiap pikselnya, yaitu bernilai 1 atau 0 [13]. Nilai 0 akan dinyatakan sebagai warna hitam sedangkan untuk nilai 1 akan dinyatakan sebagai warna putih. Jenis dari citra ini hanya membutuhkan 1-bit untuk menyimpan nilai pada setiap pikselnya. Jenis citra ini sering digunakan untuk proses *masking* ataupun proses segmentasi citra. Pada gambar 2.10 dibawah ini merupakan contoh citra biner.



Gambar 2. 10 Citra Biner

Untuk mendapatkan suatu citra biner maka kita membutuhkan citra *grayscale* yang dilakukan *thresholding* terhadapnya berdasarkan nilai ambang batas atau disebut *threshold* yang ditentukan. Jika nilai piksel pada citra *grayscale* melebihi atau menyamai nilai *threshold*, maka nilai piksel tersebut dikonversi menjadi 1. Namun jika nilai piksel kurang dari nilai *threshold* maka nilai piksel tersebut dikonversi menjadi 0. Proses konversi secara sederhana dapat ditulis menggunakan persamaan seperti berikut.

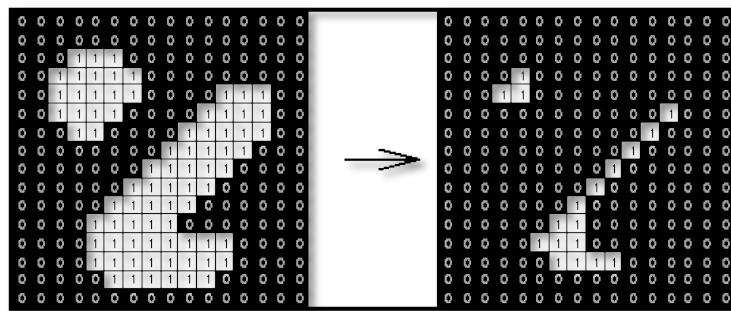
$$y' = \begin{cases} 1, & y \geq \text{threshold} \\ 0, & \text{otherwise} \end{cases} \quad (2.3)$$

2.3.3 Image Morphology

Dalam konteks pengolahan citra digital, morfologi merupakan studi mengenai topologi dan struktur objek dari sebuah citra. Proses dalam morfologi mengacu pada operasi tertentu dimana sebuah objek dalam citra akan dikenai dengan elemen penstruktur dan dengan demikian akan mengurangi atau menambah bentuk suatu objek pada citra. Pada operasi morfologi akan dibahas mengenai dua operasi yaitu erosi dan dilasi.

1. Erosion atau Erosi

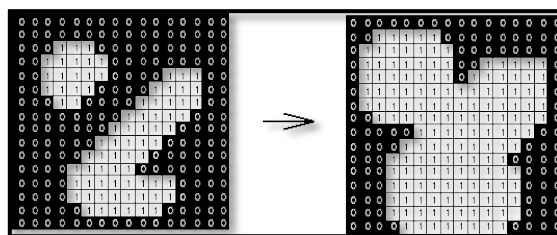
Operasi erosi adalah operasi penipisan objek yang terdapat pada citra biner. Operasi erosi dilakukan dengan cara mengurangi piksel pada kontur objek, sesuai dengan elemen penstrukturnya.



Gambar 2. 11 Proses operasi erosi

2. Dilation atau Dilasi

Operasi dilasi adalah operasi penebalan objek yang terdapat pada citra biner. Operasi dilakukan dengan menambah piksel pada kontur dari objek sesuai dengan elemen penstruktur.



Gambar 2. 12 Proses operasi dilasi

2.3.4 *Perspective Transformation* atau transformasi perspektif

Perspective transformation merupakan suatu metode dimana metode ini berfungsi untuk mengkonversi citra menjadi pandangan perspektif yang lain[14]. Ada beberapa contoh distorsi dari sebuah citra yaitu distorsi *pitch*, distorsi menyamping, dan dan distorsi campuran. Pada gambar 2.13 merupakan contoh dari beberapa macam distorsi pada citra.



Gambar 2. 13 Gambar plat yang terdistorsi[14]

2.3.5 *Background Subtraction*

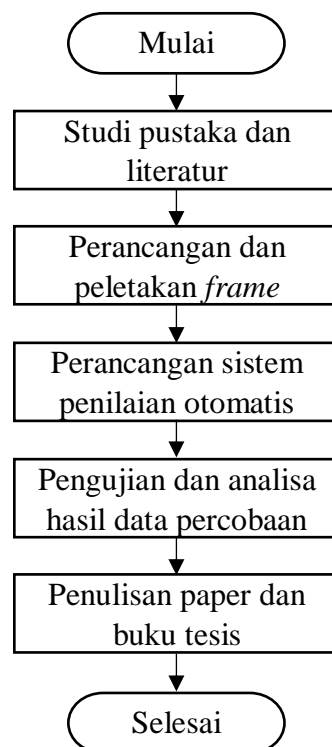
Backgoround subtraction merupakan sebuah metode yang digunakan proses komputasi untuk memisahkan objek yang berada di latar depan dengan latar belakang pada *frame* citra yang berbeda [15]. Operasi *background subtraction* banyak digunakan untuk mengurangi derau yang ada pada citra dan juga dapat mendeteksi adanya pergerakan [16].

BAB 3

METODE PENELITIAN

Pada penelitian ini berfokus pada metode yang digunakan untuk mengenali sasaran tembak dan bekas tembakan yang terdapat pada lembar sasaran tembak. Dalam penelitian ini menggunakan sebuah kamera untuk menjadi sensor dalam mendeteksi sasaran tembak dan bekas tembakan. Proses yang terjadi pada kamera yaitu untuk mendeteksi bekas tembakan pada lingkaran sasaran tembak dan akan dilakukan pendeteksian otomatis secara terus menerus hingga sesi menembak selesai.

Pada tahap ini merupakan tahap proses perancangan sistem penghitung skor tembak otomatis berbasis pemrosesan citra untuk mendeteksi dan menghitung skor tembakan yang terdapat pada sasaran tembak. Adapun tahapan dalam perancangan penelitian adalah sebagai berikut:



Gambar 3. 1 Diagram blok tahapan penelitian

3.1 Studi pustaka dan Literatur

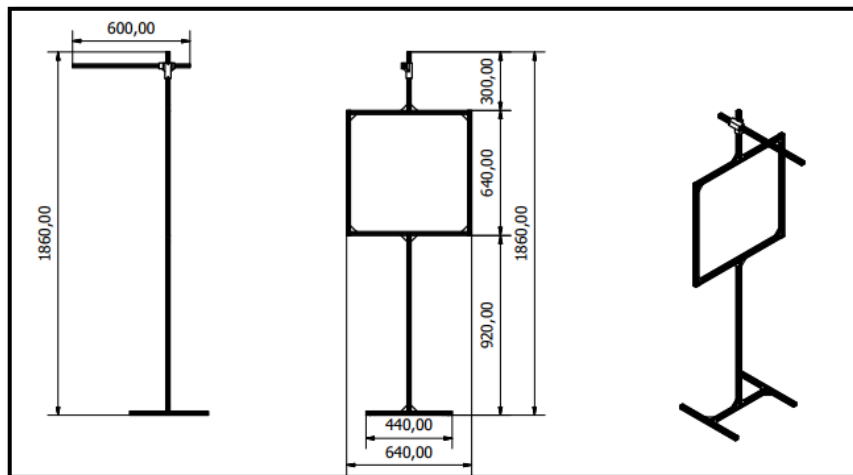
Pada tahap ini dilakukan kegiatan pengumpulan penelitian-penelitian terdahulu yang berkaitan dengan penelitian yang akan dilakukan untuk mempelajari teori-teori penunjang yang berhubungan dengan penelitian yang bersumber dari buku dan jurnal. Informasi-informasi berkaitan dengan teknik, metode dan juga analisis yang telah didapatkan akan dikaji dan diamati untuk mendapatkan metode yang tepat dalam pembuatan sistem penghitung skor tembak otomatis.

3.2 Perancangan dan Peletakan *Frame* Penghitung Skor Otomatis

Pada bab ini akan dijelaskan proses pembuatan dari *frame* penghitung skor tembak otomatis beserta penempatan *frame* saat diaplikasikan terhadap penembak saat melakukan latihan menembak.

3.2.1 Perancangan *Frame* Penghitung Skor Tembak Otomatis

Alat beserta bahan yang digunakan dalam proses pembuatan *frame* penghitung skor tembak otomatis menggunakan material yang bersifat *lowcost* sehingga dalam pembuatannya tidak membutuhkan biaya yang terlalu mahal. Material *frame* yang digunakan pada sistem penghitung skor tembak otomatis menggunakan material berupa *aluminium profile* dengan ukuran 20mm x 20mm dimana bahan tersebut dibuat untuk meletakkan lembar sasaran tembak dimana pada *frame* telah dipasang sebuah gabus berukuran 640mm x 640mm yang berfungsi untuk menempelkan lembar sasaran tembak pada *frame* dan di bagian sebelah atas depan pada *frame* telah disiapkan sebuah pegangan atau *holder* yang berfungsi untuk meletakkan kamera untuk melakukan proses pengambilan gambar terhadap lembar sasaran tembak yang telah dipasang pada *frame* untuk setiap adanya bekas tembakan yang terdapat pada sasaran tembak. Peletakan kamera pada *frame* diletakkan pada bagian sebelah atas dari *frame* untuk mengurangi adanya cahaya berlebih yang diterima oleh kamera sehingga diharapkan dapat melakukan pemrosesan citra dengan baik. *Frame* penghitung skor tembak otomatis memiliki spesifikasi ukuran tinggi sebesar 1860mm dan lebar sebesar 640 mm dimana desain dan spesifikasi ukuran *frame* secara detail dari sistem penghitung skor otomatis dapat dilihat pada gambar 3.2.



Gambar 3. 2 Desain frame skor tembak otomatis

Bentuk dari hasil pembuatan *frame* penghitung skor tembak otomatis dapat dilihat pada gambar 3.3.



Gambar 3. 3 Frame skor tembak otomatis

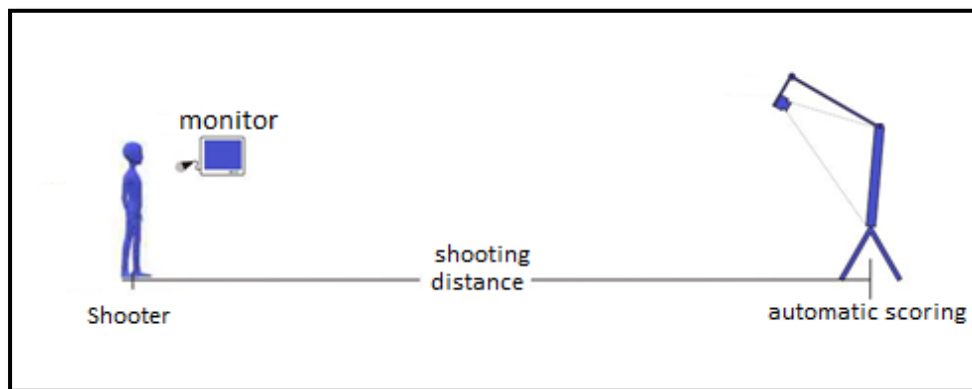
Untuk posisi peletakan kamera terhadap lembar sasaran pada *frame* penghitung skor tembak otomatis dapat dilihat pada gambar 3.4.



Gambar 3. 4 Posisi peletakan kamera pada frame

3.2.2 Peletakam *Frame* Penghitung Skor Tembak Otomatis

Pada sistem penghitung skor tembak berbasis pemrosesan citra terdapat dua jenis sistem berdasarkan penggunaan kamera yaitu sistem dengan posisi kamera yang dipasang fix pada *frame* (*Single fixed Stationary System*) dan sistem dengan posisi kamera dipasang pada senjata (*Sensor on The Gun Scoring System*). Dari dua sistem yang telah disebutkan diatas memiliki kelebihan dan kekurangan masing-masing. Pada sistem *single fixed stationary system* memiliki kelebihan gambar yang di tangkap oleh kamera lebih jelas karena kamera dipasang pada bagian *frame* sehingga jarak antara kamera terhadap sasaran tembak tidak jauh sedangkan kekurangan dari sistem ini adalah resiko kamera terkena tembakan juga tinggi sehingga dibutuhkan beberapa perlengkapan tambahan untuk melindungi kamera dan posisi kamera yang benar. Pada sistem *sensor on the gun scoring system* memiliki kelebihan yaitu kamera tidak akan beresiko terkena tembakan dikarenakan posisi kamera tidak pada *frame* namun kekurangan dari sistem ini jika penembak melakukan suatu pergerakan maka hasil dari kamera kurang jelas karena posisi antara penembak dan sasaran tembak yang cukup jauh. Pada penelitian ini menggunakan sistem *single fixed stationary system* untuk melakukan pengambilan gambar sasaran tembak. Pengambilan gambar dilakukan secara satu persatu dimana apabila ada tembakan yang tercetak pada sasaran tembak maka kamera akan mengambil gambar dengan bekas tembakan yang ada pada lembar sasaran dan akan dilakukan secara terus menerus hingga jumlah tembakan maksimal terpenuhi dimana pada penelitian ini dibatasi sebanyak sepuluh tembakan tiap lembar sasaran. Pada penelitian ini, pemrosesan citra yang dilakukan adalah pemrosesan pada citra yang sudah diambil oleh kamera sehingga gambar yang diproses merupakan gambar statis hal ini dilakukan agar pemrosesan citra tidak terlalu berat karena jika pemrosesan citra dilakukan secara langsung akan ada kemungkinan proses yang dilakukan akan sedikit lambat berhubung pemrosesan citra secara online membutuhkan proses yang berat dan memori yang memadai. Pada gambar 3.5 merupakan peletakan *frame* penghitung skor otomatis terhadap penembak beserta perlengkapan lain dalam mendukung kinerja sistem penghitung skor otomatis.

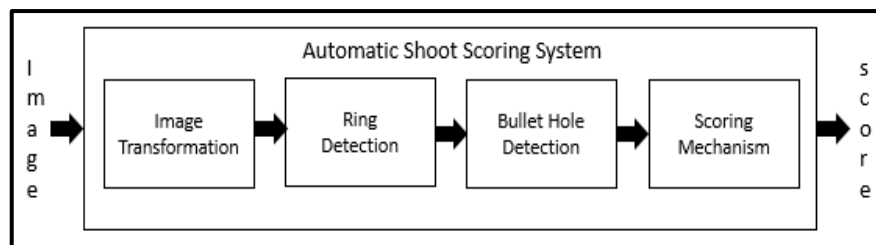


Gambar 3. 5 Peletakan frame penghitung skor otomatis terhadap penembak

3.3 Perancangan Sistem Penilaian Otomatis

Pada penelitian ini dilakukan sebuah pemrosesan citra yang bertujuan untuk melakukan proses *cropping* lembar sasaran tembak, proses transformasi citra sasaran tembak, proses pendeteksian lingkaran penilaian, proses pendeteksian citra bekas tembakan dan yang terakhir adalah proses penilaian. Dalam proses pengambilan citra sasaran tembak dengan menggunakan kamera *webcam* yang sudah dipasang pada *frame*, dari gambar yang dihasilkan oleh kamera selanjutnya dilakukan proses pendeteksian lembar sasaran tembak dengan menggunakan pendeteksian kontur pada sasaran tembak, setelah dilakukan pendeteksian sasaran tembak maka dilakukan proses *cropping* terhadap lembar sasaran tembak untuk membuang objek yang tidak diperlukan disekitar sasaran tembak. Hasil dari proses *cropping* tersebut selanjutnya dilakukan suatu proses *perspective transform* dimana proses ini bertujuan untuk mengkoreksi citra sasaran tembak yang terdistorsi karena posisi kamera terhadap sasaran tembak menghasilkan sudut sehingga menghasilkan gambar yang miring dan proses pengkoreksian citra ini dapat mempermudah dalam proses pendeteksian lingkaran penilaian pada lembar sasaran tembak. Setelah citra dilakukan proses pengkoreksian, selanjutnya dilakukan pendeteksian lingkaran untuk mendapatkan diameter dan titik pusat setiap lingkaran untuk dijadikan acuan penilaian terhadap bekas tembakan. Setelah proses pendeteksian lingkaran dilakukan, selanjutnya dilakukan proses pendeteksian bekas tembakan pada setiap citra dengan menggunakan metode *subtraction* yang berfungsi untuk mencari

perbedaan antara citra target sebelum adanya tembakan dan setelah adanya tembakan dan kemudian dicari titik tengah dari bekas tembakan untuk dilakukan proses selanjutnya yaitu penilaian terhadap lingkaran yang sudah ditentukan pada proses sebelumnya. Pemrosesan citra dilakukan secara sekuensial pada citra sasaran tembak untuk melakukan pendeteksian bekas tembakan karena pada proses pendeteksian tembakan dilakukan dengan cara membandingkan setiap citra target yang sudah diambil oleh kamera. Pada gambar 3.6 merupakan diagram blok dari sistem keseluruhan pada sistem penghitungan skor tembak otomatis berbasis pemrosesan citra.



Gambar 3. 6 Diagram blok sistem penghitung skor tembak otomatis

Berdasarkan Gambar 3.6 terdapat beberapa bagian untuk mempermudah dalam perencanaan dan pembuatan sistem penghitung skor tembak otomatis berbasis pemrosesan citra dimana dalam bab ini akan dijelaskan secara urut mengenai bagian-bagian berdasarkan blok diagram pada Gambar 3.6 dimana empat bagian tersebut adalah :

1. *Image Transformation* dimana pada proses ini bertujuan untuk mengkoreksi citra sasaran tembak yang terdistorsi karena peletakan kamera yang tidak tegak lurus terhadap sasaran tembak.
2. *Ring Detection* dimana pada proses ini bertujuan untuk mendeteksi lingkaran-lingkaran penilaian beserta diameter dan titik pusat lingkaran yang terdapat pada sasaran tembak.
3. *Bullet Hole Detection* dimana proses ini bertujuan untuk mengekstraksi bekas tembakan yang terdapat pada lembar sasaran tembak
4. *Scoring Mechanism* merupakan proses untuk melakukan penilaian berdasarkan posisi bekas tembakan terhadap lingkaran penilaian.

3.3.1 Proses *Image Transformation* pada Sasaran Tembak

Pada proses *image transformation*, pengambilan gambar sasaran tembak dilakukan dengan menggunakan sebuah *webcam* kamera *Logitech C525* kemudian dihubungkan dengan perangkat *Personal Computer* atau laptop. Pada proses pengambilan gambar dengan menggunakan kamera yang memiliki resolusi gambar 720p. Dalam proses pengambilan gambar sasaran tembak agar hasil dari pengambilan sasaran tembak memiliki hasil yang baik memerlukan pencahayaan yang cukup dimana pada proses pengambilan gambar sasaran tembak dilakukan didalam ruangan dengan pencahayaan ruangan sehingga cahaya yang terpancar berada diatas kamera. Pada Gambar 3.7 merupakan hasil dari tangkapan gambar kamera *Logitech C525*.



Gambar 3. 7 Hasil pengambilan gambar sasaran tembak

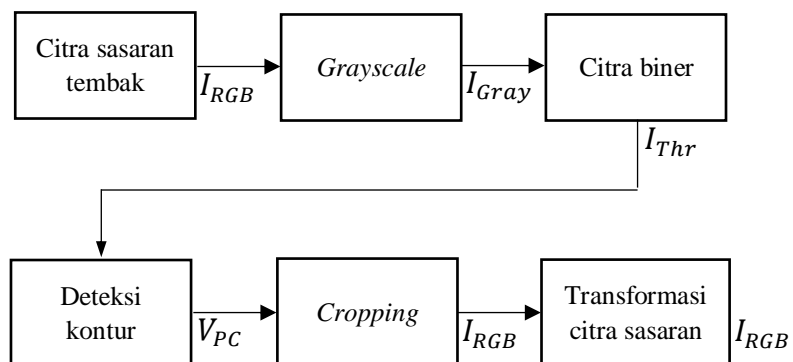
Setelah didapatkan gambar sasaran tembak seperti terlihat pada Gambar 3.7, selanjutnya dilakukan proses rotasi pada gambar sasaran tembak dengan

menggunakan fungsi rotasi yang ada pada openCV. Pada Gambar 3.8 merupakan hasil rotasi gambar sasaran tembak.



Gambar 3. 8 Gambar sasaran tembak setelah dilakukan proses rotasi

Pada proses *image transformation* terdapat beberapa proses yang harus dilakukan dimana dapat dilihat diagram blok proses *image transformation* pada gambar 3.9.



Gambar 3. 9 Blok diagram proses transformasi perspektif

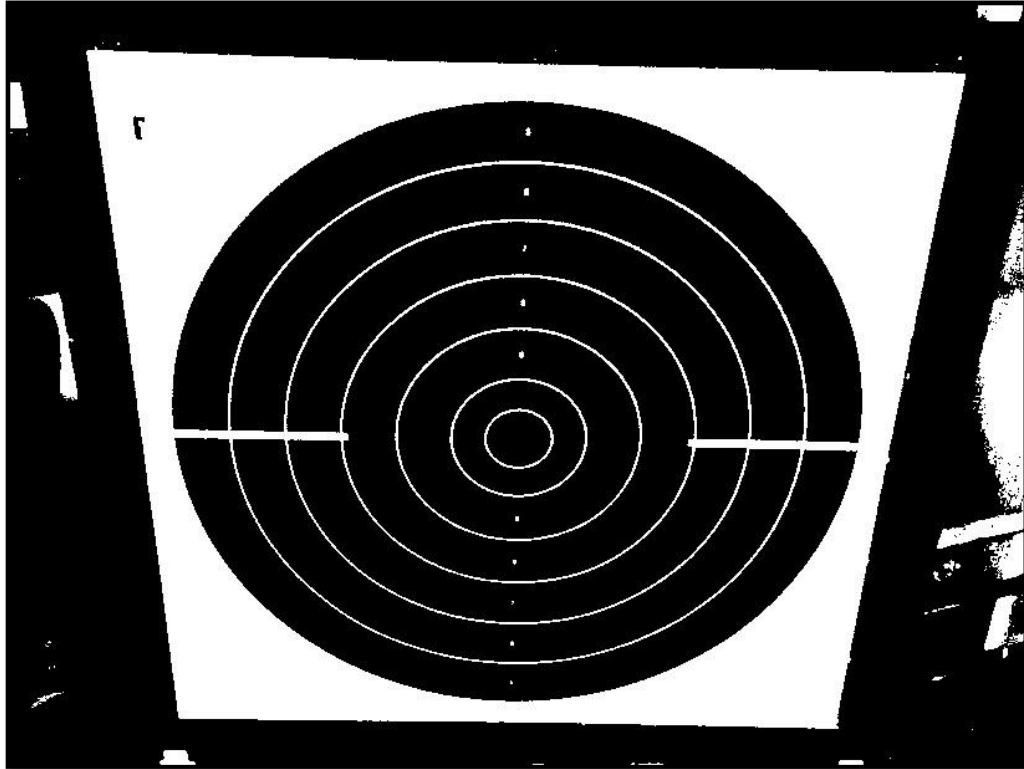
Diagram blok yang terlihat pada Gambar 3.9 memproses citra sasaran tembak yang sudah diambil oleh kamera dan dilakukan proses rotasi citra untuk kemudian dilakukan proses *grayscale* dimana tujuan dari proses *grayscale* ini untuk menjadikan citra sasaran tembak yang awalnya berupa citra berwarna dengan memiliki 3 *channel* menjadi citra keabuan. Hasil dari proses *grayscale* dapat dilihat pada Gambar 3.10.



Gambar 3. 10 Citra sasaran tembak setelah dilakukan proses *grayscale*

Pada Gambar 3.10 terlihat citra sasaran tembak tidak terdapat warna-warna lain setelah dilakukan proses *grayscale* dimana pada citra awal masih terdapat beberapa warna yang ada pada sekitar *frame* sasaran tembak. Setelah dilakukan proses *grayscale* maka proses selanjutnya adalah dengan mengaplikasikan proses *thresholding* pada citra sasaran tembak untuk mempermudah proses pendeteksian lembar sasaran tembak sebelum dilakukan proses *cropping*. Pada proses *thresholding* dilakukan dengan mengatur nilai batas *threshold* untuk menjadikan citra sasaran tembak memiliki 2 *channel* warna yaitu hitam dan putih dimana nilai *threshold* diberikan dengan cara *trial and error* untuk mendapatkan hasil yang baik.

Hasil pemrosesan citra pada proses *thresholding* pada sasaran tembak dapat dilihat pada gambar 3.11.



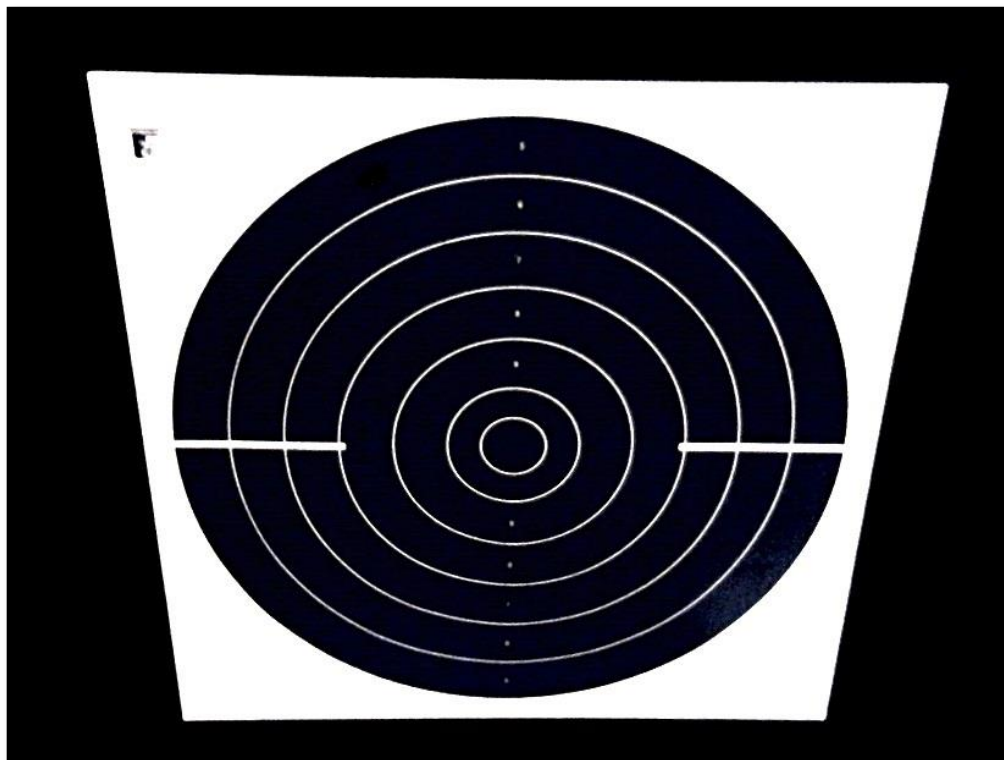
Gambar 3.11 Hasil *thresholding* pada citra sasaran tembak

Pada Gambar 3.11 terlihat hasil proses *thresholding* terhadap citra *grayscale* sebelumnya yang menghasilkan citra dengan memiliki 2 channel warna yaitu hitam dan putih untuk kemudian dilakukan proses pendeteksian kontur dengan menggunakan fungsi yang ada pada openCV untuk menemukan kontur-kontur yang ada pada citra sasaran tembak setelah dilakukan proses *thresholding* dimana tujuan dari pendeteksian kontur ini adalah untuk menemukan kontur dari lembar sasaran tembak untuk kemudian dilakukan proses *cropping* terhadap lembar sasaran tembak. Hasil dari proses pendeteksian kontur pada lembar sasaran tembak dapat dilihat pada contoh gambar 3.12 dimana kontur yang terdeteksi pada lembar sasaran tembak diberi warna hijau untuk menandakan bahwa kontur dari lembar sasaran tembak telah terdeteksi oleh fungsi pendeteksian kontur pada *library* openCV.



Gambar 3.12 Hasil pendeteksian kontur lembar sasaran tembak

Pada Gambar 3.12 merupakan hasil akhir dari pendeteksian lembar sasaran tembak dimana pada proses pendeteksian kontur yang telah dilakukan menghasilkan beberapa kontur yang terdeteksi pada citra lembar sasaran tembak sehingga untuk menentukan kontur dari sasaran tembak dengan cara menggunakan parameter *external contour* yang terdapat pada fungsi *findContours* pada openCV dan pada proses pendeteksian kontur sasaran tembak juga menggunakan batasan berupa besar area pada kontur yang terdeteksi dimana pada batas area untuk penentuan sasaran tembak diberikan nilai sebesar 300.000 untuk menentukan bahwa area dengan besar nilai tersebut merupakan area untuk lembar sasaran tembak sehingga setelah pendeteksian kontur sasaran tembak dilakukan maka kontur yang terdeteksi akan digambar pada citra rgb lembar sasaran tembak dengan menggunakan fungsi *drawContours* dengan mengisi parameter yang ada pada fungsi *drawContours*. Setelah kontur sasaran terdeteksi maka proses selanjutnya adalah menjadikan memotong atau *crop* bagian kontur yang terdeteksi yaitu pada bagian tepi sasaran tembak untuk mendapatkan citra sasaran tembak dan membuang objek lain yang berada pada daerah *frame* sasaran tembak.



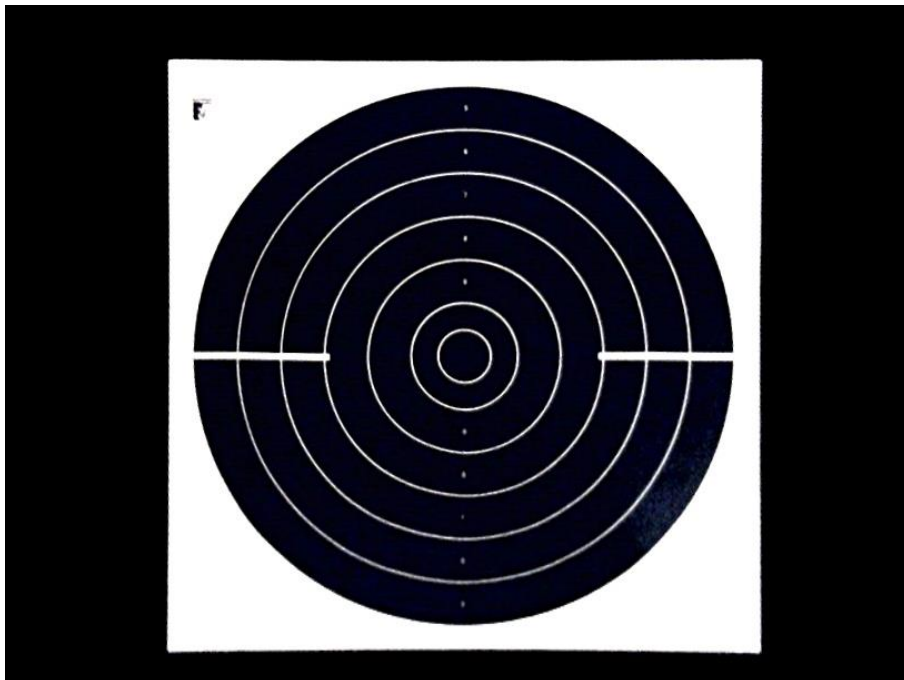
Gambar 3.13 Hasil proses *cropping* terhadap sasaran tembak

Setelah proses *cropping* dilakukan maka proses selanjutnya adalah melakukan transformasi terhadap lembar sasaran tembak yang terdistorsi untuk menjadikan lembar sasaran tembak menjadi citra yang terlihat seakan tegak lurus terhadap kamera. Pada proses *image transformation* deteksi kontur pada citra sasaran tembak dilakukan kembali namun pada pendeteksian kontur tersebut bertujuan untuk memberikan bingkai kotak pada sasaran tembak dengan menggunakan fungsi *bounding box* yang ada pada openCV dimana tujuan dari pemingkaian ini untuk dijadikan sebuah referensi transformasi terhadap citra sasaran tembak. Gambar 3.14 merupakan hasil dari pemrosesan citra sasaran tembak yang telah dilakukan proses pendeteksian kontur lembar sasaran tembak dan proses pemingkaian dimana pada gambar terlihat garis merah yang menandakan kontur dari lembar sasaran tembak dan garis kotak berwarna hijau merupakan bounding box dari lembar sasaran tembak untuk dijadikan acuan proses transformasi pada citra sasaran tembak.



Gambar 3.14 Deteksi kontur dan pemingkaian pada sasaran tembak

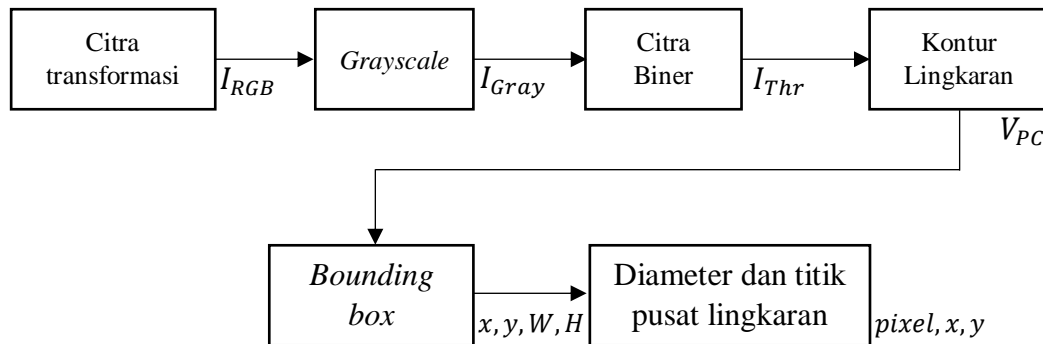
Pada Gambar 3.15 merupakan hasil dari proses transformasi citra sasaran tembak.



Gambar 3.15 Hasil transformasi citra sasaran tembak

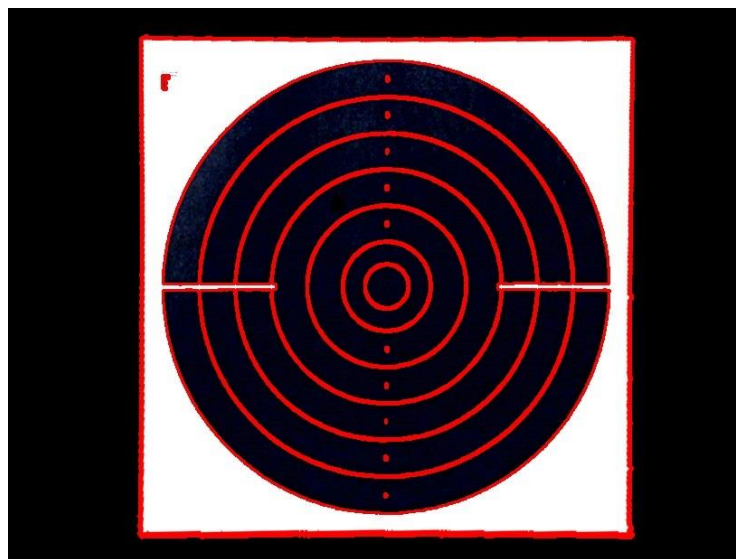
3.3.2 Proses *Ring Detection* pada Sasaran Tembak

Setelah dilakukan proses transformasi pada sasaran tembak, maka selanjutnya dilakukan proses pendeteksian lingkaran untuk menentukan diameter dan titik pusat lingkaran penilaian dalam membantu proses penilaian. Dalam proses pendeteksian lingkaran terdapat beberapa proses yang dilakukan dimana dapat dilihat dalam diagram blok pada Gambar 3.16.



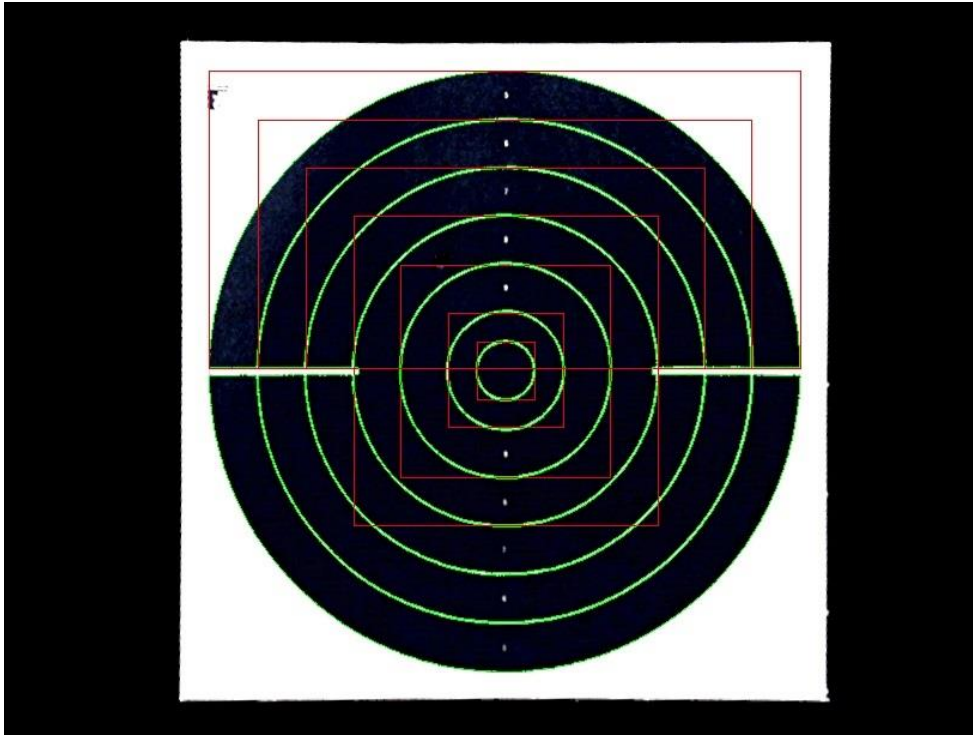
Gambar 3.16 Blok diagram pendeteksian lingkaran penilaian

Berdasarkan diagram blok pada Gambar 3.16, pendeteksian lingkaran penilaian dengan melakukan proses *grayscale* yang kemudian dilanjutkan proses *thresholding* terhadap citra sasaran tembak pada Gambar 3.15 yang telah dilakukan proses transformasi. Pada Gambar 3.17 merupakan hasil dari deteksi kontur terhadap citra sasaran tembak yang telah ditransformasi.



Gambar 3.17 Hasil deteksi kontur citra sasaran tembak

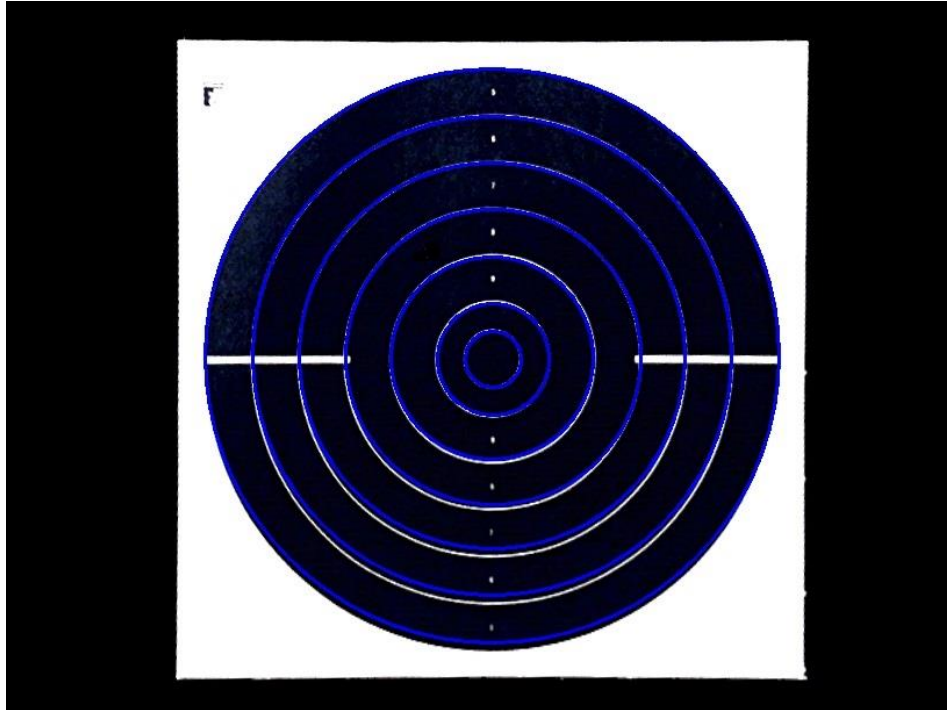
Setelah didapatkan kontur dari lembar sasaran tembak maka proses selanjutnya adalah menyeleksi kontur lingkaran penilaian dan membuang kontur yang memiliki luas area kecil sehingga *bounding box* hanya akan memberi bingkai pada kontur lingkaran penilaian saja seperti terlihat pada gambar 3.18.



Gambar 3.18 Hasil *bounding box* terhadap kontur lingkaran

Pada Gambar 3.18 terlihat kontur lingkaran diberi warna hijau dan *bounding box* diberi warna merah. Pada pengaplikasian *bounding box* terhadap lingkaran kontur terdapat tiga buah lingkaran yang terputus dikarenakan adanya garis putih yang melintang secara vertikal sehingga menjadikan kontur pada tiga buah lingkaran dari luar terputus sehingga pada proses pemingkaian terdapat *bounding box* yang sama namun pada Gambar 3.18 untuk *bounding box* dengan ukuran yang sama terhadap lingkaran terluar hanya diwakili oleh satu *bounding box* untuk mewakili satu lingkaran yang terputus. Hasil dari proses pemingkaian dengan menggunakan *bounding box* didapatkan nilai lebar bingkai pada sumbu x dan tinggi bingkai pada sumbu y sehingga dari data tersebut dapat ditentukan diameter tiap lingkaran dan titik pusat tiap lingkaran dengan membagi nilai sumbu x dan y dengan dua sehingga dapat digambarkan beberapa buah lingkaran penilaian seperti pada

Gambar 3.19 dimana lingkaran yang dihasilkan oleh proses pembingkaihan diberi warna biru.

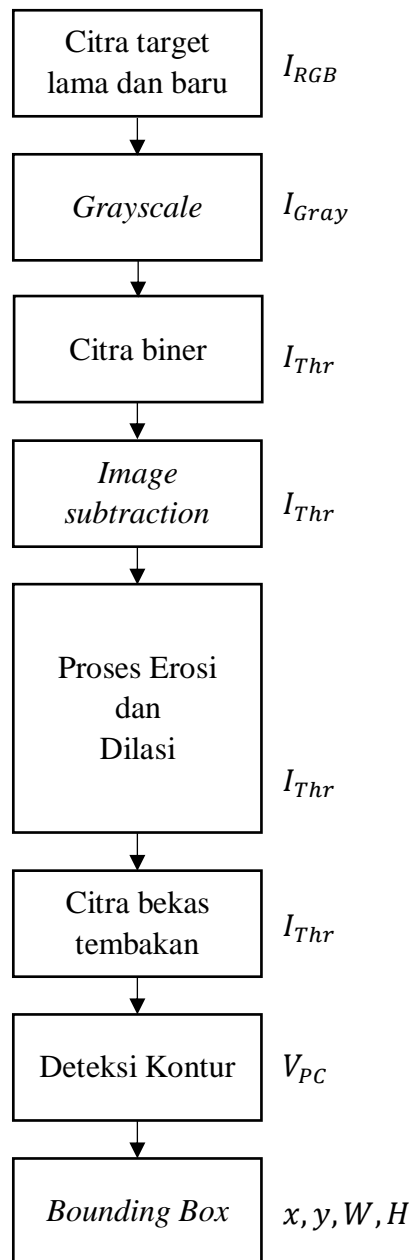


Gambar 3.19 Lingkaran penilaian hasil ekstraksi *bounding box*

Pada Gambar 3.19 telah didapatkan lingkaran penilaian untuk tiap skor tembakan dimana pada setiap lingkaran memiliki nilai diameter yang berbeda dan dijadikan kedalam radius untuk melakukan proses penilaian tembakan dimana proses penilaian tembakan dengan menghitung jarak dari posisi bekas tembakan terhadap titik pusat lingkaran dan jarak tersebut akan dibandingkan dengan nilai radius pada setiap lingkaran untuk mendapatkan skor dimana pada penilaian tersebut untuk nilai skor dimulai dari angka 5 (lima) hingga 11 (sebelas). Setelah proses pendeteksian lingkaran sasaran tembakan selesai, proses selanjutnya dengan melakukan pendeteksian bekas tembakan yang terdapat pada lembar sasaran tembak yang akan dijelaskan pada sub bab berikutnya.

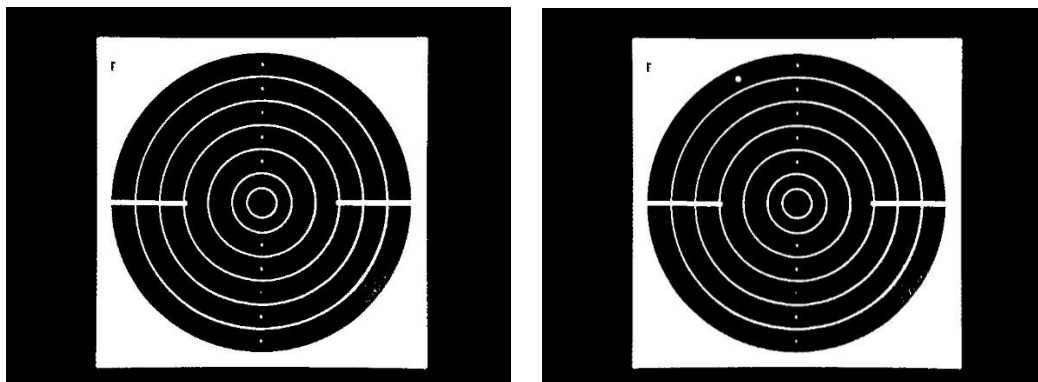
3.3.3 Proses *Bullet Hole Detection* pada Sasaran Tembak

Pada proses pendeteksian bekas tembakan, metode yang digunakan adalah *image subtraction* dimana metode tersebut merupakan metode yang sederhana, aman dan dapat memberikan hasil yang baik dalam proses pendeteksian bekas tembakan pada lembar sasaran tembak. Pada Gambar 3.20 merupakan blok diagram dalam melakukan proses *image subtraction* untuk mendeteksi adanya bekas tembakan pada lembar sasaran tembak.



Gambar 3.20 Blok diagram pendeteksian bekas tembakan

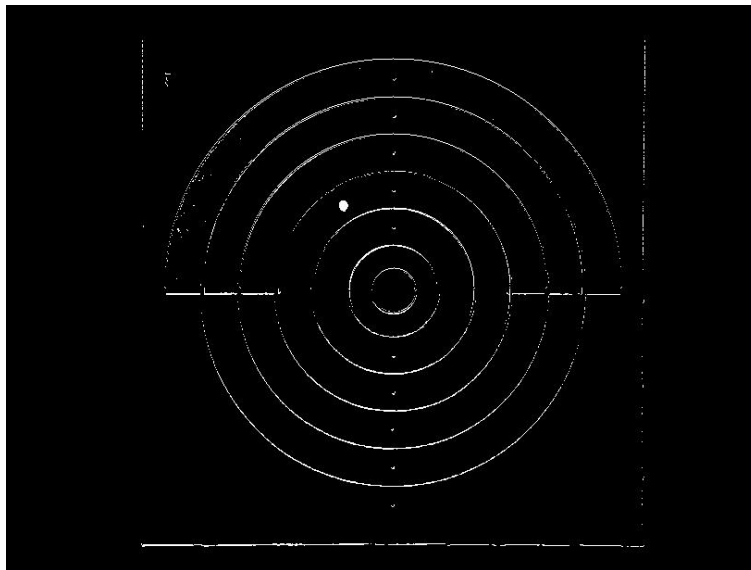
Pada penelitian ini, metode *image subtraction* sangat cocok saat diaplikasikan dalam sistem penghitung skor otomatis dimana dalam proses subtraksi dimana bertujuan untuk mengekstraksi sasaran tembak yang terdapat bekas tembakan didalamnya. Dalam proses pendeteksian bekas tembakan, cara kerja dari metode subtraksi adalah mencari adanya perbedaan pada setiap citra sasaran tembak dimana dalam proses ini akan dilakukan proses subtraksi pada citra sasaran tembak baru dengan citra sasaran tembak lama. Metode subtraksi citra yang digunakan dalam penelitian ini dapat menentukan posisi atau lokasi dari bekas tembakan yang terdapat pada sasaran tembak yang telah diambil oleh kamera. Setiap gambar sasaran tembak yang diambil oleh kamera akan disimpan sebagai data acuan untuk melakukan proses subtraksi citra pada setiap citra sasaran baru dengan bekas tembakan yang baru pada setiap citra sasaran tembak. Proses subtraksi citra diawali dengan melakukan proses *grayscale* lalu kemudian diikuti proses *thresholding* pada citra sasaran tembak lama dan citra sasaran tembak baru. Setelah dihasilkan citra biner pada sasaran tembak lama dan sasaran tembak baru, maka proses selanjutnya dengan melakukan proses subtraksi pada citra biner. Gambar 3.21 merupakan gambar dari proses *thresholding* pada citra sasaran tembak lama dan citra sasaran tembak baru.



(a) Citra sebelum ada tembakan (b) Citra setelah ada tembakan

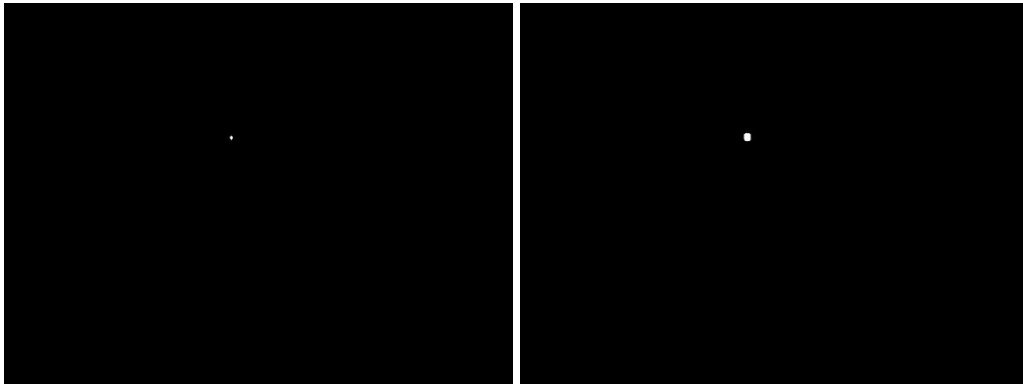
Gambar 3. 21 Perbedaan citra sasaran tembak

Pada gambar 3.21 terlihat dimana gambar (a) belum terdapat adanya bekas tembakan sedangkan pada gambar (b) terdapat satu bekas tembakan dimana terletak pada lingkaran paling luar. Setelah didapatkan dua buah gambar yang berbeda maka proses selanjutnya dengan mengaplikasikan subtraksi citra pada kedua gambar tersebut sehingga hasil dari subtraksi citra pada kedua gambar tersebut dapat dilihat pada Gambar 3.22.



Gambar 3. 22 Hasil subtraksi citra sasaran tembak

Hasil subtraksi citra pada Gambar 3.22 terlihat memiliki *noise* berupa garis-garis lingkaran dan sisi dari lembar sasaran tembak dimana *noise* ini dapat mengakibatkan proses pendeteksian bekas tembakan terganggu. Hal ini terjadi karena sifat dari metode subtraksi memiliki kelemahan dimana metode subtraksi citra sangat sensitif terhadap perubahan kecil sehingga dapat mengakibatkan adanya *noise* pada hasil subtraksi citra dan dengan adanya kejadian seperti hal tersebut dilakukan sebuah proses filter morfologi dimana dengan melakukan proses erosi dan dilasi pada citra subtraksi. Proses erosi berfungsi untuk menipiskan objek pada citra biner dengan mengurangi piksel dari kontur objek yang dapat menghilangkan *noise* pada citra biner namun hal ini menjadikan citra bekas tembakan menjadi kecil sehingga dilakukan penebalan piksel dengan menggunakan proses dilasi dimana hasil dari proses filter morfologi terlihat pada Gambar 3.23.

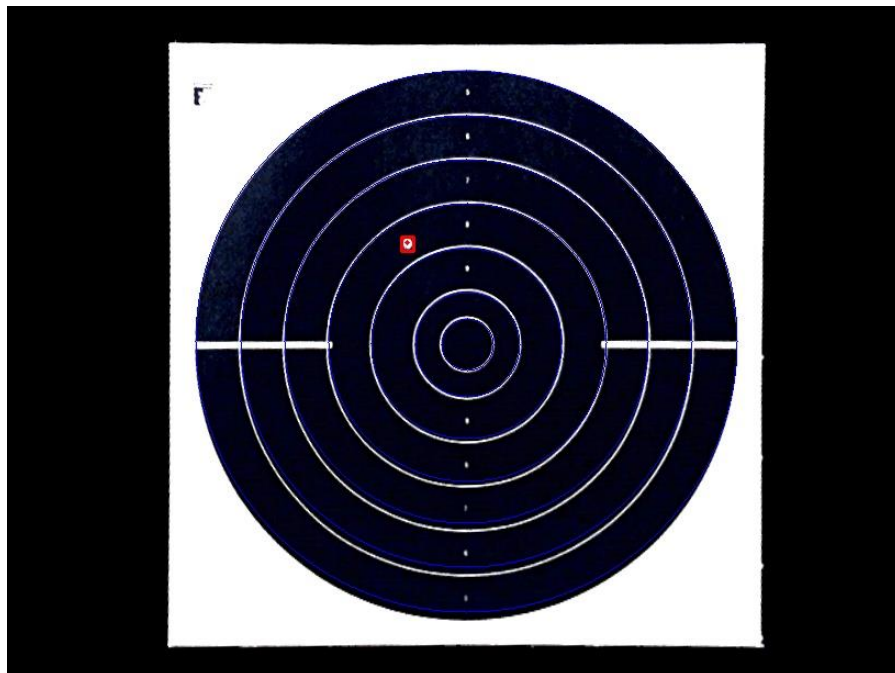


(a) Hasil proses erosi

(b) Hasil proses dilasi

Gambar 3. 23 Hasil proses filter morfologi bekas tembakan

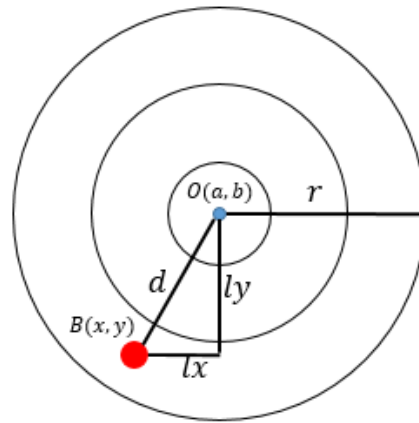
Langkah selanjutnya setelah melakukan proses subtraksi citra yaitu mendeteksi kontur dari citra sasaran tembak untuk kemudian diberikan bingkai kotak atau *bounding box* untuk mendapatkan nilai panjang pada sumbu x dan tinggi pada sumbu y dari bekas tembakan untuk kemudian mendapatkan titik tengah dari bekas tembakan yang terdeteksi. Gambar 3.24 merupakan hasil dari proses pembingkaiian pada citra bekas tembak



Gambar 3. 24 Bekas tembak terdeteksi dengan diberi bingkai merah

3.3.4 Scoring Mechanism

Beberapa proses yang dilakukan sebelumnya telah mendapatkan data diameter dan titik pusat pada setiap lingkaran penilaian dan mampu mendeteksi posisi dari setiap bekas tembakan yang terdapat pada lembar sasaran tembak sehingga dari data yang didapatkan pada proses tersebut dapat digunakan sebagai data untuk penilaian tembakan. Penilaian dilakukan dengan cara menghitung jarak dari titik pusat bekas tembakan menuju titik pusat lingkaran penilaian dimana proses dari penentuan jarak dapat dilakukan dengan menggunakan rumus *Euclidean Distance*. Pada Gambar 3.25 merupakan representasi dari posisi bekas tembakan pada lingkaran penilaian.



Gambar 3. 25 Representasi posisi bekas tembakan

Pada gambar terlihat beberapa *variabel* untuk melakukan proses penilaian dimana pada gambar terdapat lingkaran merah yang merepresentasikan bekas tembakan (B) yang terletak pada koordinat x dan y, lingkaran biru (O) yang merepresentasikan pusat lingkaran penilaian, r merupakan nilai radius yang dimiliki setiap lingkaran penilaian, d merupakan jarak dari pusat tembakan menuju pusat lingkaran, lx merupakan jarak posisi tembakan dari pusat tembakan menuju pusat lingkaran pada sumbu x, dan ly merupakan jarak dari pusat tembakan menuju pusat lingkaran pada sumbu y. Rumus dari *euclidean distance* dalam proses penilaian dapat dilihat pada persamaan (1).

$$d(B, O) = \sqrt{(a - x)^2 + (b - y)^2} \quad (3.1)$$

BAB 4

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan mengenai pengujian dan analisa terhadap proses penilaian tembakan dengan menggunakan stiker lingkaran berdiameter 7.62 mm dan dalam percobaan dilakukan secara *offline* dengan mengambil sepuluh set data simulasi tembakan dimana pada tiap set data simulasi memiliki sepuluh citra sekuensial lubang tembakan.

4.1 Pengujian penilaian simulasi tembakan menggunakan stiker

4.1.1 Pengujian Simulasi Tembakan set Ke-1

Tabel 4.1 Hasil Percobaan Simulasi Set Ke-1

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	8	8
2	7	7
3	11	11
4	10	10
5	8	8
6	5	5
7	6	6
8	6	7
9	9	9
10	7	7

4.1.2 Pengujian Simulasi Tembakan set Ke-2

Tabel 4.2 Hasil Percobaan Simulasi Set Ke-2

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	11	0
2	11	5
3	9	9
4	8	6
5	6	0
6	7	5
7	8	8
8	8	8
9	9	0
10	9	9

4.1.3 Pengujian Simulasi Tembakan Set Ke-3

Tabel 4.3 Hasil percobaan simulasi set Ke-3

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	8	8
2	10	10
3	9	9
4	9	9
5	8	8
6	9	9
7	8	8
8	11	11
9	11	0
10	10	5

4.1.4 Pengujian Simulasi Tembakan set Ke-4

Tabel 4.4 Hasil percobaan simulasi set Ke-4

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	6	6
2	7	7
3	8	8
4	9	9
5	10	10
6	9	9
7	9	9
8	11	11
9	11	11
10	11	11

4.1.5 Pengujian Simulasi Tembakan Set Ke-5

Tabel 4.5 Hasil percobaan simulasi set Ke-5

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	6	6
2	7	7
3	10	10
4	10	10
5	11	11
6	9	9
7	10	10
8	9	9
9	5	5
10	10	10

4.1.6 Pengujian Simulasi Tembakan Set Ke-6

Tabel 4.6 Hasil percobaan simulasi set Ke-6

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	10	10
2	10	10
3	9	9
4	7	7
5	8	8
6	8	8
7	9	9
8	8	8
9	9	9
10	11	11

4.1.7 Pengujian Simulasi Tembakan Set Ke-7

Tabel 4.7 Hasil percobaan simulasi set Ke-7

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	7	7
2	9	9
3	11	11
4	11	11
5	11	11
6	10	10
7	8	8
8	9	9
9	9	9
10	9	9

4.1.8 Pengujian Simulasi Tembakan Sset Ke-8

Tabel 4.8 Hasil percobaan simulasi set Ke-8

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	9	9
2	5	5
3	7	7
4	8	8
5	10	10
6	9	9
7	10	10
8	11	11
9	10	10
10	8	8

4.1.9 Pengujian Simulasi Tembakan Set Ke-9

Tabel 4.9 Hasil percobaan simulasi set Ke-9

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	10	10
2	10	10
3	11	11
4	9	9
5	10	10
6	11	11
7	9	9
8	10	10
9	9	9
10	9	9

4.1.10 Pengujian Simulasi Tembakan Set Ke-10

Tabel 4.10 Hasil percobaan simulasi set Ke-10

Tembakan Ke-	Skor Simulasi	Skor Didapatkan
1	5	5
2	8	8
3	9	9
4	10	10
5	11	11
6	9	9
7	8	8
8	10	10
9	8	8
10	10	10

4.1.11 Analisa Data Percobaan

Pada pengujian simulasi tembakan telah dilakukan proses penilaian pada 10 set citra dimana setiap set citra memiliki data citra tembakan berjumlah 100 citra bekas tembakan. Pada percobaan yang telah dilakukan terdapat 9 data citra yang terjadi kesalahan dalam proses penilaian bekas tembakan dari 100 citra yang diujikan dalam proses *scoring* dimana kesalahan tersebut dikarenakan pada saat proses subtraksi citra terhadap setiap lembar sasaran tembak, proses erosi dari hasil subtraksi yang telah dilakukan kurang baik dan masih terdapat beberapa *noise* sehingga dengan terjadinya hal tersebut menjadikan *noise* dari hasil morfologi dianggap sebagai citra bekas tembakan. Dengan terjadinya kesalahan tersebut, proses penilaian yang dilakukan pada bekas tembakan terhadap lingkaran penilaian kurang baik dan skor yang dihasilkan bukan merupakan skor dari bekas tembakan melainkan *noise* dari hasil subtraksi citra lembar sasaran tembak. Pada percobaan ini, tingkat akurasi yang diperoleh dengan cara simulasi memiliki persentase sebesar 91%.

Halaman ini sengaja dikosongkan

BAB 5

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dari penelitian yang telah dilakukan, sistem penghitung skor tembak otomatis yang telah dibuat dapat melakukan pendeteksian dan penilaian terhadap lubang tembakan dengan persentase akurasi sebesar 91% dengan menggunakan stiker lingkaran berdiameter 7.62mm sebagai simulasi lubang bekas tembakan. Pengujian dilakukan dengan menggunakan data citra tembakan sebanyak 100 data dimana terdapat 9 data citra yang terjadi kesalahan dalam proses penilaian yang disebabkan oleh *noise* yang masih tersisa setelah proses subtraksi citra dilakukan.

5.2 Saran

Setelah melakukan percobaan penilaian otomatis dengan mensimulasikan lubang bekas tembakan dengan menggunakan kertas stiker, ada beberapa saran untuk penelitian selanjutnya yaitu untuk lubang bekas tembakan menggunakan citra lubang tembakan asli dari bekas tembakan senjata. Selain hal tersebut, diharapkan pada penelitian selanjutnya terdapat tampilan yang dapat menunjang kinerja dari sistem penghitung skor tembak otomatis sehingga dapat menampilkan informasi tembakan dengan lebih baik.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] (online) <https://www.britannica.com/sports/shooting> [diakses pada 4:35 11-12-2017]
- [2] K.S. Hatamleh, Q.A. Khasawneh, L. Sawaqed, M.M. Hasan, S. Yafawi, M. Al-Shabi. 2015. Evolutionary Low Cost Visual Shooting Practice System. International Symposium on Mechatronics and its Applications (ISMA).
- [3] Aryan, P. Ruswono. 2012. Vision based Automatic Target Scoring System for Mobile Shooting Range. International Conference on Advanced Computer Science and Information Systems (ICACSIS).
- [4] Xinnan Fan, Qianqian Cheng, Penghua Ding, Xuewu Zhang. 2009. Design of Automatic Target Scoring System of Shooting Game based on Computer Vision. IEEE International Conference on Automation and Logistics.
- [5] Cuiliu Ye, Hong Mi. 2011. The Technology of Image Processing used in Automatic Target Scoring System. Fourth International Joint Conference on Computational Sciences and Optimization.
- [6] Y.C. Lin, S.G. Miaou, Y. Cheng Lin, S.L. Chen. 2015. An Automatic Scoring System for Air Pistol Shooting Competition Based on Image Recognition of Target Sheets. International Conference on Consumer Electronics-Taiwan (ICCE-TW).
- [7] H. Harmaen. 2012. Design and Implementation Automatic Scoring Computation System for Shooting Sport with Template Matching. INKOM, Universitas Bina Nusantara, Jakarta, Indonesia.
- [8] S. Ying, W. Gang, and W. Yaojun. 2011. The Application of Information Technology in Sports Training. International Conference on Future Computer Science and Education.
- [9] Soetedjo, A., Mahmudi, A., Ashari, M.I., Nakhoda, Y.I. 2014. Implementation of Sensor on the Gun System Using Embedded Camera for Shooting Training. International Conference on Technology, Informatics, Management, Engineering and Environment Bandung, Indonesia.
- [10] <http://www.issf-sports.org/theissf/history.ashx> [diakses pada 5:11 11-12-2017]
- [11] https://en.wikipedia.org/wiki/Shooting_target [diakses pada 6:16 11-12-2017]
- [12] <https://en.wikipedia.org/wiki/Camera> [diakses pada 13:34 14-12-2017]
- [13] Priyanto Hidayatullah. 2017. Pengolahan Citra Digital Teori dan Aplikasi Nyata. Penerbit INFORMATIKA
- [14] Baoyang, Jesse S. Jin, Fei Li, Xianfeng Han, Wei Tong, Mingjie Wang. 2016. A Perspective Correction Method Based On The Bounding Rectangle and Least Square Fitting. International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP).
- [15] Shahrizat Shaik Mohamed, Nooritawati Md Tahir, Ramli Adnan. 2010. Background Modelling and Background Subtraction Performance for Object Detection. International Colloquium on Signal Processing and Its Application (ICSPA).

- [16]. Imam Mustakim. 2011. Sistem Pengenalan Wajah dengan Metode Subspace LDA (Linear Discriminant Analysis). Tugas Akhir Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember Surabaya.

LAMPIRAN

Program Pemrosesan Citra

```
#include "opencv2\opencv.hpp"
#include "opencv2\core\core.hpp"
#include "opencv2\highgui\highgui.hpp"
#include "opencv2\imgproc\imgproc.hpp"
#include <array>
#include <algorithm>
#include "circle_detection.cpp"
#include <conio.h>
```

```
using namespace cv;
using namespace std;
```

```
#define MAX_SHOOT 10
```

```
int x_bullet_coord[MAX_SHOOT];
int y_bullet_coord[MAX_SHOOT];
int status_index_shoot[MAX_SHOOT];
int radius_shoot[MAX_SHOOT];
int score_shoot[MAX_SHOOT];
int index_shoot = 0;
int x_global[7];
int y_global[7];
int fl = 0;
```

```
int x_center_bullet;
int y_center_bullet;
```

```
Point center_global;
```

```
Mat img_rgb,
    img_gry,
    img_bin,
    img_crp_gry,
```

```

    img_crp_bin,
    trf_gry,
    trf_bin,
    gauss_blur;

vector<Vec4i> hierarchy;
vector<Point> approx_circle;
vector<vector<Point>> contours;
vector<vector<Point>> rect_contours;
vector<vector<Point>> circ_contours;
vector<vector<Point>> poly_contours(1);

int largest_cnt_idx;
int largest_area = 0;
int largest_circle_area = 0;
int largest_circle_index = 0;
int x_bullet, y_bullet;
int score_accumulate=0;

double srt_area[20];
int a;
int radius_score[7];

double    area,
          circ_area;

//fungsi untuk mengurutkan rectangle berdasarkan
kontur
vector<Rect> urut(vector<Rect> dt)
{
    Rect temp;
    for (int i=0;i<dt.size();i++)
    {
        for (int j=i;j<dt.size();j++)
        {
            if(dt[i].width>dt[j].width && i!=j)

```

```

        {
            temp=dt[i];
            dt[i]=dt[j];
            dt[j]=temp;
        }
    }
    return dt;
}
/*
Mat image_transform(Mat img_rgb)
{
    Point2f img_rgb_center(img_rgb.cols/2.0F,
img_rgb.rows/2.0F);
    Mat rotated_img =
getRotationMatrix2D(img_rgb_center, 180, 1);
    warpAffine(img_rgb, img_rgb, rotated_img,
img_rgb.size());
    //imshow("My Thesis RGBrot", img_rgb);
    cvtColor(img_rgb, img_gry, CV_BGR2GRAY);

    //imshow("My Thesis GRY", img_gry);
    threshold(img_gry, img_bin, 40, 255,
CV_THRESH_BINARY);
    //imshow("My Thesis BIN", img_bin);
    findContours(img_bin, rect_contours,
CV_RETR_EXTERNAL, CV_CHAIN_APPROX_SIMPLE);
    Mat drawing = Mat::zeros(img_rgb.size(),
CV_8UC1);

    for(int i = 0; i < rect_contours.size(); i++)
    {
        area = contourArea(rect_contours[i]);
        //printf("kontur ke-%d dan luas kontur = %f
\n", i, area);
        if(area > 300000)

```

```

        {drawContours(drawing, rect_contours, i,
CV_RGB(255,255,255), CV_FILLED);}
    }

    Mat img_crp = Mat::zeros(img_rgb.rows,
img_rgb.cols, CV_8UC3);
    img_crp.setTo(Scalar(0,0,0));
    img_rgb.copyTo(img_crp, drawing);
    normalize(drawing.clone(), drawing, 0.0, 255.0,
CV_MINMAX, CV_8UC1);
    GaussianBlur(img_crp, gauss_blur, cv::Size(0,0),
3);
    addWeighted(img_crp, 2, gauss_blur, -0.5, 0,
img_crp);
    //imshow("My Thesis CRP RGB", img_crp);
    cvtColor(img_crp, img_crp_gry, CV_BGR2GRAY);
    //imshow("My
Thesis CRP GRY", img_crp_gry);
    threshold(img_crp_gry, img_crp_bin, 40, 255,
CV_THRESH_BINARY);
    //imshow("My Thesis CRP BIN", img_crp_bin);
    Mat img_dst(img_crp.rows, img_crp.cols, CV_8UC1,
CV_RGB(0,0,0));
    findContours(img_crp_bin, contours,
CV_RETR_EXTERNAL, CV_CHAIN_APPROX_SIMPLE);

    for(int i = 0; i < contours.size(); i++)
    {
        area = contourArea(contours[i], false);
        if(area > largest_area){largest_area = area;
largest_cnt_idx = i;}
    }
    approxPolyDP(Mat(contours[largest_cnt_idx]),
poly_contours[0], 5, true);
    //printf("Segi terdeteksi : %d \n",
poly_contours[0].size());

```

```

        Rect boundrect =
boundingRect(contours[largest_cnt_idx]);
        if(poly_contours[0].size() >= 4)
        {
            vector<Point2f> quad_pts;
            vector<Point2f> square_pts;

            quad_pts.push_back(Point2f(poly_contours[0][0].x,
poly_contours[0][0].y));

            quad_pts.push_back(Point2f(poly_contours[0][1].x,
poly_contours[0][1].y));

            quad_pts.push_back(Point2f(poly_contours[0][3].x,
poly_contours[0][3].y));

            quad_pts.push_back(Point2f(poly_contours[0][2].x,
poly_contours[0][2].y));

            square_pts.push_back(Point2f(boundrect.x +
85,
            boundrect.y));
            square_pts.push_back(Point2f(boundrect.x +
85,
            boundrect.y +
boundrect.height));
            square_pts.push_back(Point2f(boundrect.x +
boundrect.width-90, boundrect.y));
            square_pts.push_back(Point2f(boundrect.x +
boundrect.width-90, boundrect.y + boundrect.height));
            Mat transmatx =
getPerspectiveTransform(quad_pts, square_pts);
            Mat transform = Mat::zeros(img_crp.rows,
img_crp.cols, CV_8UC3);
            warpPerspective(img_crp, transform,
transmatx, img_crp.size());
            return transform;

```

```

    }
    return img_rgb;
}
*/

ring_detection My_Thesis(Mat img_rgb)
{
    ring_detection point;
    int x_center_circle = 0;
    int y_center_circle = 0;

    //imshow("My Thesis RGB", img_rgb);
    Point2f img_rgb_center(img_rgb.cols / 2.0F,
img_rgb.rows / 2.0F);
    //program untuk store variabel pada sumbu x dan y
float
    Mat rotated_img =
getRotationMatrix2D(img_rgb_center, 180, 1);
//program untuk melakukan
proses rotasi matrix 2 dimensi
    warpAffine(img_rgb, img_rgb, rotated_img,
img_rgb.size());
    //program untuk
    //imshow("My Thesis RGBrot", img_rgb);
    //imwrite("(dok paper)citra rgb
awal.jpg",img_rgb);
    cvtColor(img_rgb, img_gry, CV_BGR2GRAY);

    //program untuk konversi citra rgb -> gray
    //imshow("My Thesis GRY", img_gry);
    //imwrite("citra grayscale.jpg",img_gry);
    threshold(img_gry, img_bin, 60, 255,
CV_THRESH_BINARY);
//program untuk thresholding gray ->
citra biner

```



```

        //imshow("My Thesis BIN", img_bin);
        //imwrite("citra biner.jpg",img_bin);
        findContours(img_bin, rect_contours,
CV_RETR_EXTERNAL, CV_CHAIN_APPROX_SIMPLE);
        //program untuk mendeteksi kontur pada citra
biner
        Mat drawing = Mat::zeros(img_rgb.size(),
CV_8UC1);
        //program untuk membuat frame baru dengan
ukuran yang sama dengan citra awal dengan 1 channel
warna
        for(int i = 0; i < rect_contours.size(); i++)
        {
            area = contourArea(rect_contours[i]);

            //program untuk mendeteksi besar area pada kontur
yang terdeteksi
            //printf("kontur ke-%d dan luas kontur = %f
\n", i, area);
            //drawContours(img_rgb, rect_contours, i,
CV_RGB(0,255,0), 2, 8);
            if(area > 300000)

            //program untuk menentukan besar batas area
pada citra
            {
                drawContours(drawing, rect_contours, i,
CV_RGB(255,255,255), CV_FILLED);
                //program untuk menggambar kontur yang terdeteksi
pada frame drawing
                //drawContours(img_rgb, rect_contours,
i, CV_RGB(255,0,0), 2, 8);
                //drawContours(img_bin, rect_contours,
i, CV_RGB(0,255,0), 2, 8);
            }
        }
    }

```

```

    imshow("deteksi kontur", drawing);
    //imwrite("(dok paper)citra kontur crop
awal.jpg",drawing);
    //imshow("deteksi kontur", img_rgb);
    //imwrite("kontur terdeteksi.jpg", img_rgb);
    //imwrite("(dok paper)citra kontur crop
awal.jpg",drawing);
    Mat img_crp = Mat::zeros(img_rgb.rows,
img_rgb.cols, CV_8UC3);
        //program untuk membuat frame baru dengan
ukuran sebesar img_rgb dengan 3 channel warna
        img_crp.setTo(Scalar(0,0,0));

    //program untuk membuat frame img_crp berwarna
hitam
    img_rgb.copyTo(img_crp, drawing);

    //program untuk menyalin drawing pada img_crp
    normalize(drawing.clone(), drawing, 0.0, 255.0,
CV_MINMAX, CV_8UC1);
    //program untuk
    GaussianBlur(img_crp, gauss_blur, cv::Size(0,0),
3);
    addWeighted(img_crp, 2, gauss_blur, -0.5, 0,
img_crp);
    //imshow("My Thesis CRP RGB", img_crp);
    //imwrite("crop.jpg",img_crp);
    cvtColor(img_crp, img_crp_gry, CV_BGR2GRAY);

    //program untuk mengkonversi citra rgb -> gray
    //imshow("My Thesis CRP GRY", img_crp_gry);
    threshold(img_crp_gry, img_crp_bin, 100, 255,
CV_THRESH_BINARY);
    //program untuk mengkonversi citra gray -> biner
    //imshow("My Thesis CRP BIN", img_crp_bin);

```

```

    Mat img_dst(img_crp.rows, img_crp.cols, CV_8UC1,
CV_RGB(0,0,0));
    //program untuk membuat frame baru dengan ukuran
frame sama dengan ukuran img_crp dengan 1 channel
warna

    findContours(img_crp_bin, contours,
CV_RETR_EXTERNAL, CV_CHAIN_APPROX_SIMPLE);
    //program untuk mendeteksi kontur pada citra
biner
    for(int i = 0; i < contours.size(); i++)
    {
        area = contourArea(contours[i], false);
        if(area > largest_area){largest_area = area;
largest_cnt_idx = i;}
        //drawContours(img_crp, contours, i,
CV_RGB(255,0,0), 2,8);
        //program untuk menggambar kontur pada citra
crop dengan garis warna merah
    }
    //imshow("Deteksi Kontur pada Citra Crop",
img_crp);
    //imwrite("(Dokumen Buku) Kontur terdeteksi.jpg",
img_crp);

    approxPolyDP(Mat(contours[largest_cnt_idx]),
poly_contours[0], 5, true);
    //program untuk melakukan pendeteksian bentuk
kotak dengan pendekatan segi terdeteksi
    //printf("Segi terdeteksi : %d \n",
poly_contours[0].size());
    Rect boundrect =
boundingRect(contours[largest_cnt_idx]);
    //program untuk
melakukan bounding box pada citra kontur yang
terdeteksi

```

```

    if(poly_contours[0].size() >= 4)

        //pendeteksian segi dengan jumlah lebih dari sama
        dengan 4 untuk bentuk persegi
        {
            vector<Point2f> quad_pts;
            vector<Point2f> square_pts;

            quad_pts.push_back(Point2f(poly_contours[0][0].x,
poly_contours[0][0].y));

            quad_pts.push_back(Point2f(poly_contours[0][1].x,
poly_contours[0][1].y));

            quad_pts.push_back(Point2f(poly_contours[0][3].x,
poly_contours[0][3].y));

            quad_pts.push_back(Point2f(poly_contours[0][2].x,
poly_contours[0][2].y));

            square_pts.push_back(Point2f(boundrect.x +
85,
            boundrect.y));
            square_pts.push_back(Point2f(boundrect.x +
85,
            boundrect.y +
boundrect.height));
            square_pts.push_back(Point2f(boundrect.x +
boundrect.width-75, boundrect.y));
            square_pts.push_back(Point2f(boundrect.x +
boundrect.width-75, boundrect.y + boundrect.height));

            Mat transmatx =
getPerspectiveTransform(quad_pts, square_pts);
            Mat transform = Mat::zeros(img_crp.rows,
img_crp.cols, CV_8UC3);

```

```

        warpPerspective(img_crp, transform,
transmatx, img_crp.size());

        Point P1 = poly_contours[0][0];
        Point P2 = poly_contours[0][1];
        Point P3 = poly_contours[0][2];
        Point P4 = poly_contours[0][3];

        line(img_rgb, P1, P2, CV_RGB(0, 255, 0), 1,
CV_AA, 0);
        line(img_rgb, P2, P3, CV_RGB(0, 255, 0), 1,
CV_AA, 0);
        line(img_rgb, P3, P4, CV_RGB(0, 255, 0), 1,
CV_AA, 0);
        line(img_rgb, P4, P1, CV_RGB(0, 255, 0), 1,
CV_AA, 0);

        //rectangle(img_crp,boundrect,Scalar(0,0,255),2,8
,0);
        //imshow("boundrect rgb", img_crp);
        //imwrite("boundrect.jpg",img_crp);

        //rectangle(transform,boundrect,Scalar(0,255,0),1
,8,0);
        //imshow("My Thesis TRF", transform);

        //imwrite("transform.jpg",transform);

        point.trans_img=transform;

        cvtColor(transform, trf_gry, CV_BGR2GRAY);

        //program untuk konversi citra transformasi rgb -
> gray

```

```

        //imshow("My Thesis TRF GRY", trf_gry);
        threshold(trf_gry, trf_bin, 60, 255,
CV_THRESH_BINARY);
        //program untuk konversi citra transformasi
gray -> biner
        //imshow("My Thesis TRF BIN", trf_bin);
        //imwrite("transform binary.jpg",trf_bin);

        findContours(trf_bin, circ_contours,
hierarchy, CV_RETR_CCOMP, CV_CHAIN_APPROX_SIMPLE);
        sort(circ_contours.begin(),
circ_contours.end(), [](const vector<Point> contour1,
const vector<Point> contour2)
        {return contourArea(contour1, false) <
contourArea(contour2, false);});

        vector<vector<Point> > contours_poly (
circ_contours.size() );
        vector<Rect> boundRect (
circ_contours.size() );
        vector<Point2f> center
( circ_contours.size() );
        vector<float> radius
( circ_contours.size() );

        for( int i = 0; i < circ_contours.size();
i++ )
        {
            approxPolyDP( Mat(circ_contours[i]),
contours_poly[i], 3, true );
            boundRect[i] = boundingRect(
Mat(contours_poly[i]) );
            minEnclosingCircle(
(Mat)contours_poly[i], center[i], radius[i] );
            //drawContours(transform,
circ_contours, i, CV_RGB(255,0,0), 2,8);

```

```

    }
    //imshow("kontur lingkaran", transform);
    //imwrite("kontur lingkaran.jpg",transform);

    int temp = 999;
    boundRect= urut(boundRect);
    int j = 0;
    for(int i = 0; i < circ_contours.size();
i++)
    {
        circ_area =
        contourArea(circ_contours[i], false);
        //printf("besar area %f \n",
        circ_area);
        if (circ_area > 150)
        {
            approxPolyDP(circ_contours[i],
            approx_circle,
            arcLength(Mat(circ_contours[i]),true)*0.01, true);
            if(approx_circle.size() > 4)
            {
                if(temp-boundRect[i].y>5)
                {
                    temp=boundRect[i].y;
                    //rectangle( transform,
                    boundRect[i].tl(), boundRect[i].br(), CV_RGB(0,255,0),
                    2, 8, 0 );

                    int
                    x=boundRect[i].x+boundRect[i].width/2;
                    int
                    y=boundRect[i].y+boundRect[i].width/2;

                    //circle(transform,
                    Point2f(x,y), 1,
                    CV_RGB(255,0,0),-1, 8, 0);    //titik tengah

```

```

                                circle(transform,
Point2f(x,y), boundRect[i].width/2,
                                CV_RGB(0,0,255), 1, 8, 0);    //lingkaran ke-n

                                radius_score[j] =
boundRect[i].width/2;

                                /*if(y > 0)
{
                                x_global[f1] = x;
                                y_global[f1] = y;
                                printf("Scoring = %d
\t besar radius = %d \t koord_x = %d \t koord_y =
%d\n", j,
boundRect[i].width/2,x_global[0],y_global[0]);
                                f1++;
                                //                                printf("koor x : %d
dan koor y : %d \n\n", x_global, y_global);
                                }*/

                                //printf("Scoring = %d \t
besar radius = %d \t koord_x = %d \t koord_y = %d\n",
j, boundRect[i].width/2,x_global[f1],y_global[f1]);
                                //printf("Scoring = %d \t
besar radius = %d \t koord_x = %d \t koord_y = %d \n",
j+1, boundRect[i].width/2,x,y);
                                x_center_circle += x;
                                y_center_circle += y;
                                point.circle_radius[j] =
boundRect[i].width/2;

                                j++;
                                }
                                }
                                }
                                x_center_circle/=j;

```



```

        y_center_circle/=j;
        //printf("x koor mean : %d \t y koor mean :
%d \n", x_center_circle, y_center_circle);
        imshow("My Thesis Circle Detection",
transform);
        //citra target setelah deteksi lingkaran
        //imwrite("circ detection.jpg",transform);
    }
    point.circle_coordinate = Point(x_center_circle,
y_center_circle);
    return point;
}

```

```

Point My_Image_Subtraction(Mat img_sv_prv, Mat
img_sv_nxt)

```

```

{
    Mat img_sv_prv_gry, img_sv_prv_bin,
img_sv_prv_bil,
        img_sv_nxt_gry, img_sv_nxt_bin,
img_sv_nxt_bil,
        img_sv_sbt, img_sv_sbt_erode,
img_sv_sbt_dilate, img_sv_sbt_bil;

```

```

    Mat bilateral;

```

```

    //img_sv_prv = imread("sv_im0_skor0.jpg",1);

```

```

    //imshow("Image Prev",img_sv_prv);
    imwrite("prev.jpg", img_sv_prv);
    cvtColor(img_sv_prv, img_sv_prv_gry,
CV_BGR2GRAY);
    //imshow("Image Prev Gray", img_sv_prv_gry);
    threshold(img_sv_prv_gry, img_sv_prv_bin, 40,
255, CV_THRESH_BINARY);
    //imshow("Image Prev Binary", img_sv_prv_bin);
    //imwrite("Image Prev.jpg", img_sv_prv_bin);

```

```

//img_sv_nxt = imread("sv_im1_skor5.jpg",1);

//imshow("Image Next",img_sv_nxt);
imwrite("next.jpg", img_sv_nxt);
cvtColor(img_sv_nxt, img_sv_nxt_gry,
CV_BGR2GRAY);
//imshow("Image Next Gray", img_sv_nxt_gry);
threshold(img_sv_nxt_gry, img_sv_nxt_bin, 40,
255, CV_THRESH_BINARY);
//imshow("Image Next Binary", img_sv_nxt_bin);
//imwrite("Image Next.jpg", img_sv_nxt_bin);

//absdiff(img_sv_prv_bin, img_sv_nxt_bin,
img_sv_sbt);
//imshow("Image Subtraction", img_sv_sbt);
subtract(img_sv_nxt_bin, img_sv_prv_bin,
img_sv_sbt);
imshow("Image Subtraction", img_sv_sbt);
//imwrite("Image Subtraction.jpg", img_sv_sbt);
erode(img_sv_sbt, img_sv_sbt_erode, Mat(),
Point(-1,-1), 1, 1, 1);
imshow("Image Subtraction erode",
img_sv_sbt_erode);
//imwrite("Erode.jpg", img_sv_sbt_erode);
dilate(img_sv_sbt_erode, img_sv_sbt_dilate,
Mat(), Point(-1,-1), 3, 1, 1);
imshow("Image Subtraction dilate",
img_sv_sbt_dilate);
//imwrite("Dilate.jpg", img_sv_sbt_dilate);

vector<vector<Point>> contours;

findContours(img_sv_sbt_dilate, contours,
CV_RETR_EXTERNAL, CV_CHAIN_APPROX_SIMPLE);

```

```

        vector<vector<Point>>
contours_poly(contours.size());
        vector<Rect> boundRect (contours.size());

        for(int i = 0; i < contours.size(); i++)
        {
            circ_area = contourArea(contours[i], false);
            if (circ_area > 50){
                approxPolyDP(Mat(contours[i]),
contours_poly[i], 3, true);
                boundRect[i] =
boundingRect(Mat(contours_poly[i]));
                //drawContours(img_sv_nxt,
contours_poly, i, CV_RGB(255,0,0), 1, 8);
                rectangle(img_sv_nxt,
boundRect[i].tl(), boundRect[i].br(), CV_RGB(255,0,0),
2, 8);

                //rectangle(img_sv_sbt_dilate,
boundRect[i].tl(), boundRect[i].br(), CV_RGB(255,0,0),
2, 8);

                x_center_bullet =
boundRect[i].x+boundRect[i].width/2;
                y_center_bullet =
boundRect[i].y+boundRect[i].width/2;

                circle(img_sv_nxt,
Point2f(x_center_bullet, y_center_bullet), 2,
CV_RGB(255,0,0), -1, 8, 0);
                //printf("x_center_bullet : %d \t
y_center_bullet : %d \n", x_center_bullet,
y_center_bullet);
            }
        }
        imshow("Bullet Contours", img_sv_nxt);
        //imshow("Box Contours", img_sv_sbt_dilate);

```

```

        //imwrite("Percobaan Set.jpg", img_sv_nxt);
        return Point(x_center_bullet, y_center_bullet);
    }

```

```

int main()
{

```

```

    /*string camera[] = {
        "new0.jpg",
        "new1.jpg",
        "new2.jpg",
        "new3.jpg",
        "new4.jpg",
        "new5.jpg",
        "new6.jpg",
        "new7.jpg",
        "new8.jpg",
        "new9.jpg",
        "new10.jpg"
    };*/

```

```

    /*string camera[] = {
        "coba0.jpg",
        "coba1.jpg",
        "coba2.jpg",
        "coba3.jpg",
        "coba4.jpg",
        "coba5.jpg",
        "coba6.jpg",
        "coba7.jpg",
        "coba8.jpg",
        "coba9.jpg",
        "coba10.jpg"
    };*/

```

```

//////////////////////////////////// Data Stiker
////////////////////////////////////

```

```

/*
    string camera[] = { "im0_skor0.jpg",
        "im1_skor5.jpg", "im2_skor8.jpg", "im3_skor11.jpg",
        "im4_skor9.jpg", "im5_skor9.jpg",
            "im6_skor8.jpg",
        "im7_skor10.jpg", "im8_skor8.jpg", "im9_skor11.jpg",
        "im10_skor7.jpg"

                                };

*/

//////////////////////////////////// Data Stiker
////////////////////////////////////

/*string camera[] = {    "Pict1.jpg", "Pict2.jpg",
    "Pict3.jpg", "Pict4.jpg", "Pict5.jpg",
                                "Pict6.jpg", "Pict7.jpg",
    "Pict8.jpg", "Pict9.jpg", "Pict10.jpg", "Pict11.jpg"
                                };*/

/*string camera[] = {    "pic0_scr0.jpg",
    "pic1_scr9.jpg", "pic2_scr8.jpg", "pic3_scr10.jpg",
    "pic4_scr9.jpg",
                                "pic5_scr7.jpg",
    "pic6_scr8.jpg", "pic7_scr11.jpg", "pic8_scr7.jpg",
    "pic9_scr9.jpg", "pic10_scr6.jpg"
                                };*/

// Data Offline Stiker Thesis/---
//data stiker 1
    string camera[] = { "Picture 1.jpg", "Picture
2.jpg", "Picture 3.jpg", "Picture 4.jpg", "Picture
5.jpg",

```

```

        "Picture 6.jpg", "Picture
7.jpg", "Picture 8.jpg", "Picture 9.jpg", "Picture
10.jpg", "Picture 11.jpg"
    };

    //data stiker 2
    /*string camera[] = {    "Picture 12.jpg",
"Picture 13.jpg", "Picture 14.jpg", "Picture 15.jpg",
"Picture 16.jpg",
        "Picture 17.jpg", "Picture
18.jpg", "Picture 19.jpg", "Picture 20.jpg", "Picture
21.jpg", "Picture 22.jpg"
    };*/

    //data stiker 3
    /*string camera[] = {    "Picture 23.jpg",
"Picture 24.jpg", "Picture 25.jpg", "Picture 26.jpg",
"Picture 27.jpg",
        "Picture 28.jpg", "Picture
29.jpg", "Picture 30.jpg", "Picture 31.jpg", "Picture
32.jpg", "Picture 33.jpg"
    };*/

    //data stiker 4
    /*string camera[] = {    "Picture 34.jpg",
"Picture 35.jpg", "Picture 36.jpg", "Picture 37.jpg",
"Picture 38.jpg",
        "Picture 39.jpg", "Picture
40.jpg", "Picture 41.jpg", "Picture 42.jpg", "Picture
43.jpg", "Picture 44.jpg"
    };*/

    //data stiker 5
    /*string camera[] = {    "Picture 45.jpg",
"Picture 46.jpg", "Picture 47.jpg", "Picture 48.jpg",
"Picture 49.jpg",

```

```
        "Picture 50.jpg", "Picture  
51.jpg", "Picture 52.jpg", "Picture 53.jpg", "Picture  
54.jpg", "Picture 55.jpg"
```

```
    };*/
```

```
    //data stiker 6
```

```
    /*string camera[] = {    "Picture 123.jpg",  
"Picture 124.jpg", "Picture 125.jpg", "Picture  
126.jpg", "Picture 127.jpg",  
        "Picture 128.jpg", "Picture  
129.jpg", "Picture 130.jpg", "Picture 131.jpg",  
"Picture 132.jpg", "Picture 133.jpg"
```

```
    };*/
```

```
    //data stiker 7 //error proses morfologi membuat  
peluru tidak terdeteksi
```

```
    /*string camera[] = {    "Picture 134.jpg",  
"Picture 135.jpg", "Picture 136.jpg", "Picture  
137.jpg", "Picture 138.jpg",  
        "Picture 139.jpg", "Picture  
140.jpg", "Picture 141.jpg", "Picture 142.jpg",  
"Picture 143.jpg", "Picture 144.jpg"
```

```
    };*/
```

```
    //data stiker 8          //skor 10 ada error
```

```
    /*string camera[] = { "Picture 145.jpg", "Picture  
146.jpg", "Picture 147.jpg", "Picture 148.jpg",  
"Picture 149.jpg",  
        "Picture 150.jpg", "Picture  
151.jpg", "Picture 152.jpg", "Picture 153.jpg",  
"Picture 154.jpg", "Picture 155.jpg"
```

```
    };*/
```

```
    //data stiker 9
```

```

        /*string camera[] = { "Picture a.jpg", "Picture
b.jpg", "Picture c.jpg", "Picture d.jpg", "Picture
e.jpg",
                                "Picture f.jpg", "Picture
g.jpg", "Picture h.jpg", "Picture i.jpg", "Picture
j.jpg", "Picture k.jpg"
                                };*/

//data stiker 10    //error peluru ke-6 skor 9
/*string camera[] = {    "Picture
163.jpg", "Picture 164.jpg", "Picture 165.jpg",
"Picture 166.jpg", "Picture 167.jpg",
                                "Picture 168.jpg", "Picture
169.jpg", "Picture 170.jpg", "Picture 171.jpg",
"Picture 172.jpg", "Picture 173.jpg"
                                };*/

//Data Intensitas 29 lux
/*string camera[] = {    "29a.jpg", "29b.jpg",
"29c.jpg", "29d.jpg", "29e.jpg",
                                "29f.jpg", "29g.jpg",
"29h.jpg", "29i.jpg", "29j.jpg", "29k.jpg"
                                };*/

//Data Intensitas 29 lux

//Data Intensitas 44 lux
/*string camera[] = {    "44a.jpg", "44b.jpg",
"44c.jpg", "44d.jpg", "44e.jpg",
                                "44f.jpg", "44g.jpg",
"44h.jpg", "44i.jpg", "44j.jpg", "44k.jpg"
                                };*/

//Data Intensitas 44 lux

//Data Intensitas 68 lux
/*string camera[] = {    "68a.jpg", "68b.jpg",
"68c.jpg", "68d.jpg", "68e.jpg",

```



```

        "68f.jpg", "68g.jpg",
"68h.jpg", "68i.jpg", "68j.jpg", "68k.jpg"
        };*/

//Data Intensitas 68 lux
// Data Offline Stiker Thesis//

//New Data

Mat img_rgb = imread(camera[0], 1);
ring_detection circle_center =
My_Thesis(img_rgb);
Mat temp_rgb = circle_center.trans_img;

for(int cam = 1; cam < 12; cam++)
{
    Mat img_rgb = imread(camera[cam], 1);
    circle_center = My_Thesis(img_rgb);
    img_rgb = circle_center.trans_img;
    Point scoring =
My_Image_Subtraction(temp_rgb, img_rgb);
    int radius_scoring =
        sqrt(
            pow((double)(scoring.x -
circle_center.circle_coordinate.x), 2) +
            pow((double)(scoring.y -
circle_center.circle_coordinate.y), 2)
        );
    int ring_score = 0;
    int shot_score = 0;
    for(int i = 7; i > 0; i--)
    {
        if(circle_center.circle_radius[i-
1]>radius_scoring) ring_score=i;
        if(ring_score == 7) shot_score = 5;
        if(ring_score == 6) shot_score = 6;
    }
}

```

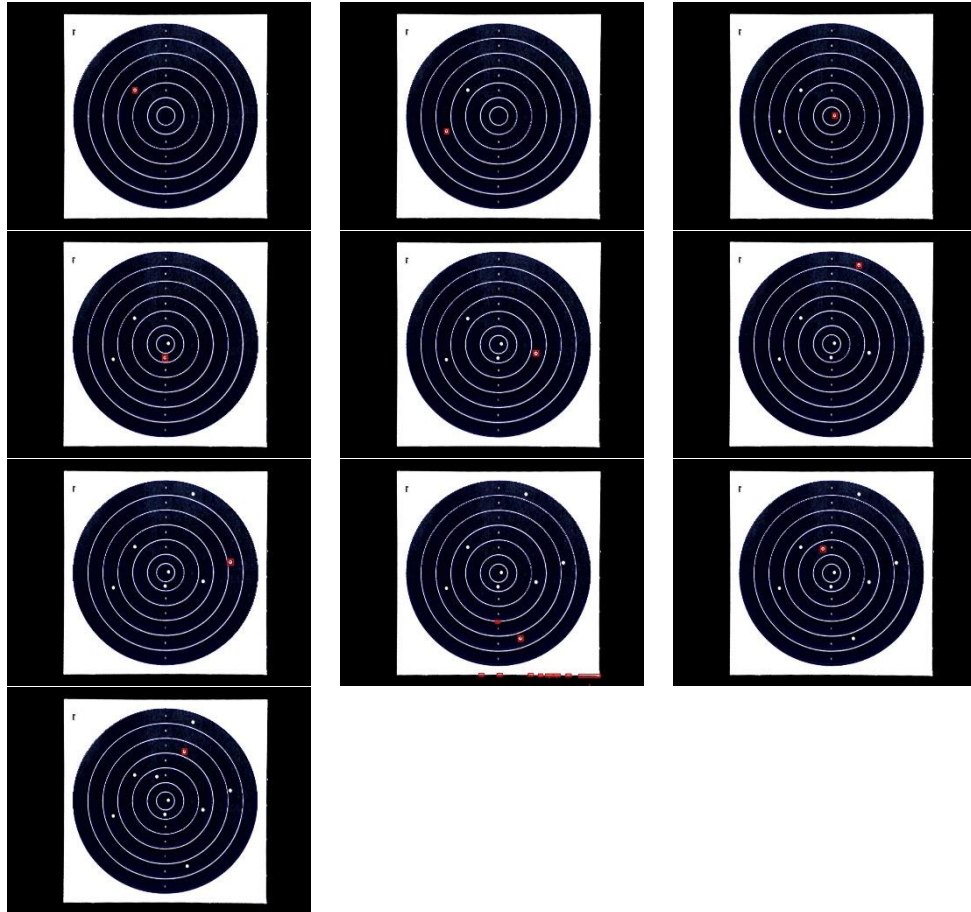
```

        if(ring_score == 5) shot_score = 7;
        if(ring_score == 4) shot_score = 8;
        if(ring_score == 3) shot_score = 9;
        if(ring_score == 2) shot_score = 10;
        if(ring_score == 1) shot_score = 11;
    }
    score_accumulate += shot_score;
    printf("Radius Scoring = %d, Ring Score = %d, Shot Score = %d, Total Score = %d\n\n",
        radius_scoring,
        ring_score,
        shot_score,
        score_accumulate);
    temp_rgb = img_rgb;
    waitkey(50);
    getch();
}
waitkey(0);
return 0;
}

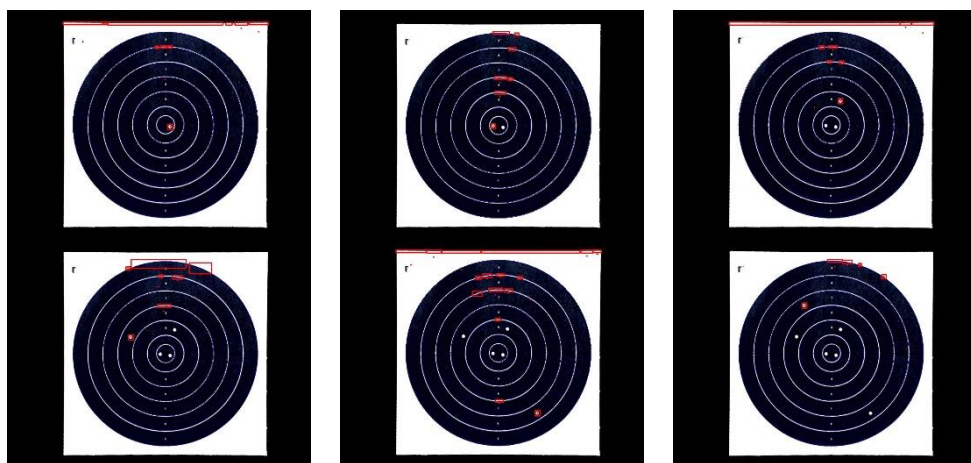
```

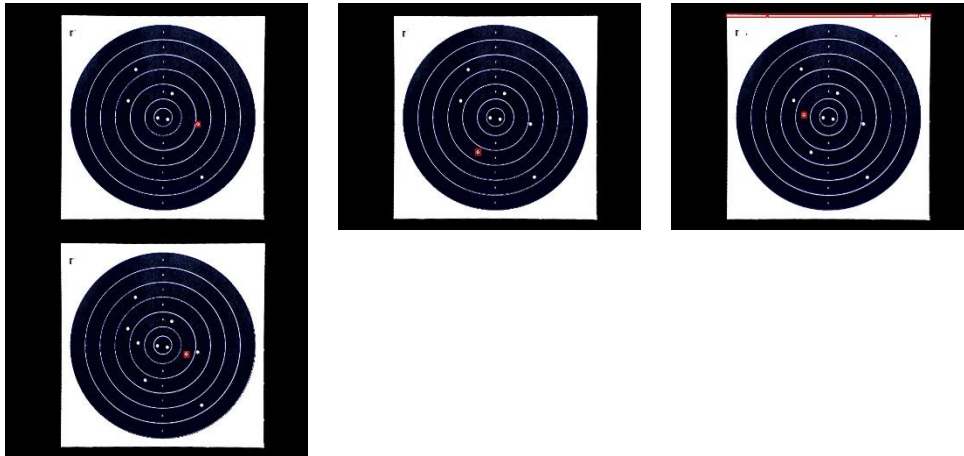
Citra Data Tembakan Menggunakan Stiker

Data Ke-1

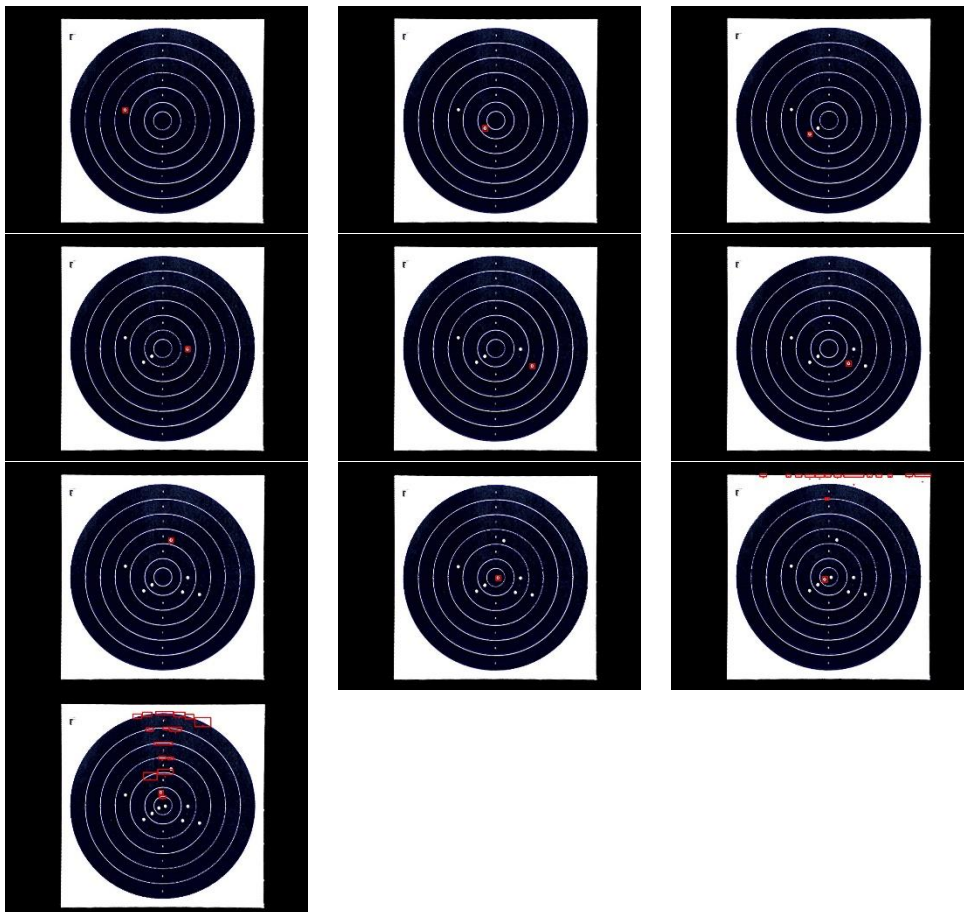


Data Ke-2

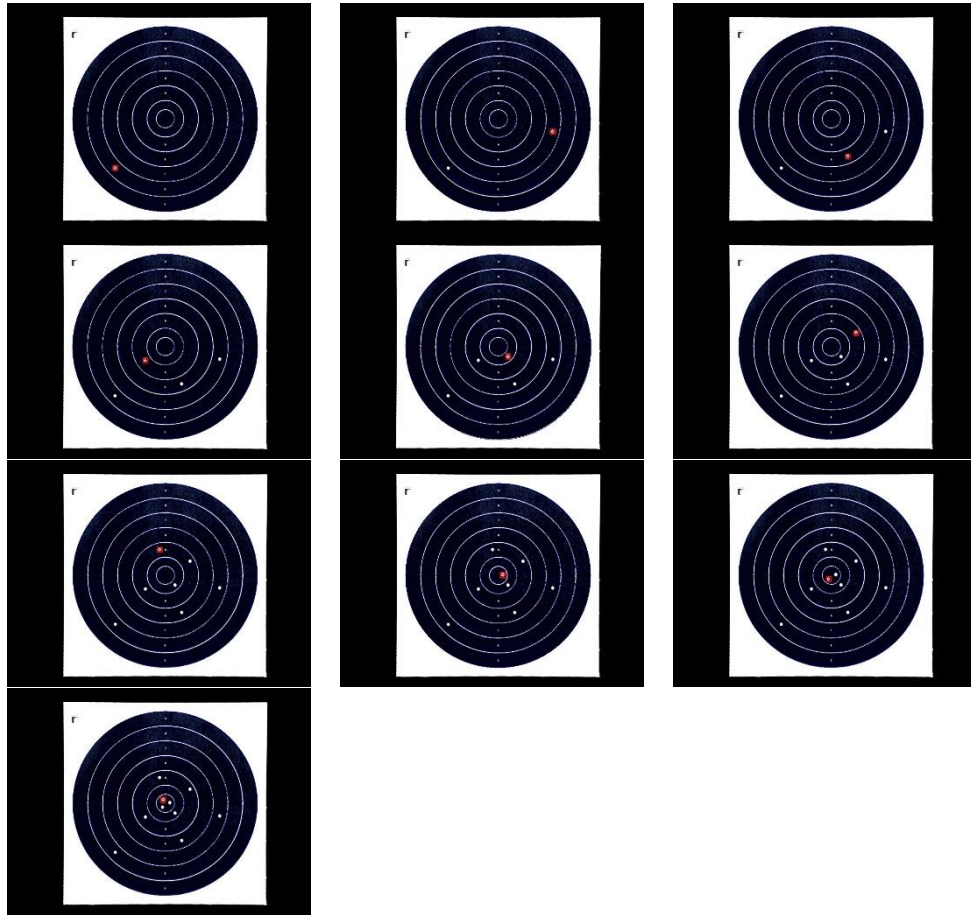




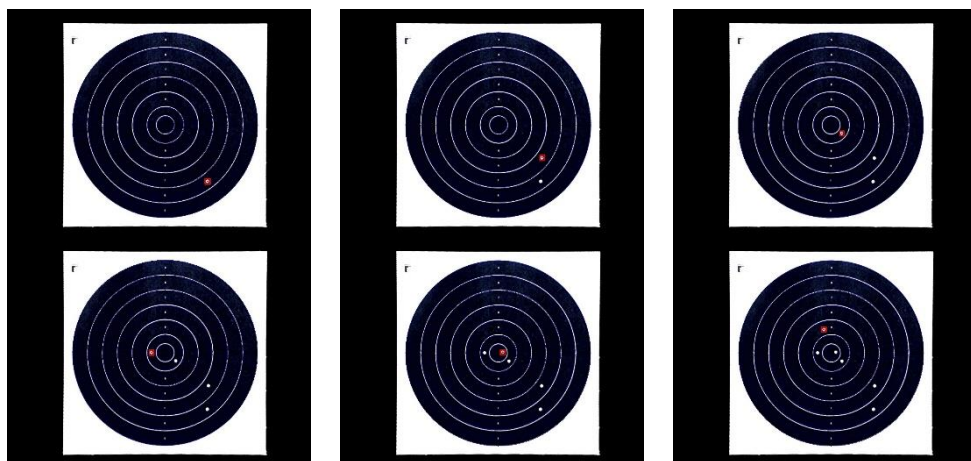
Data Ke-3

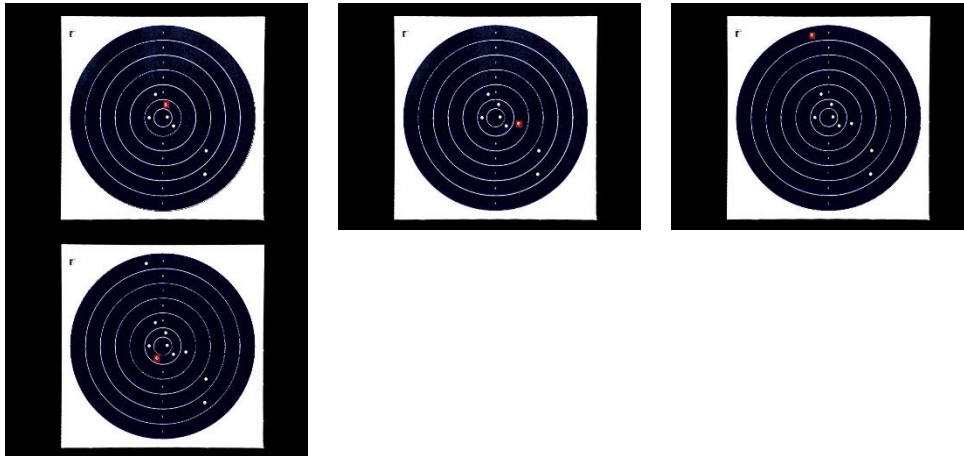


Data Ke-4

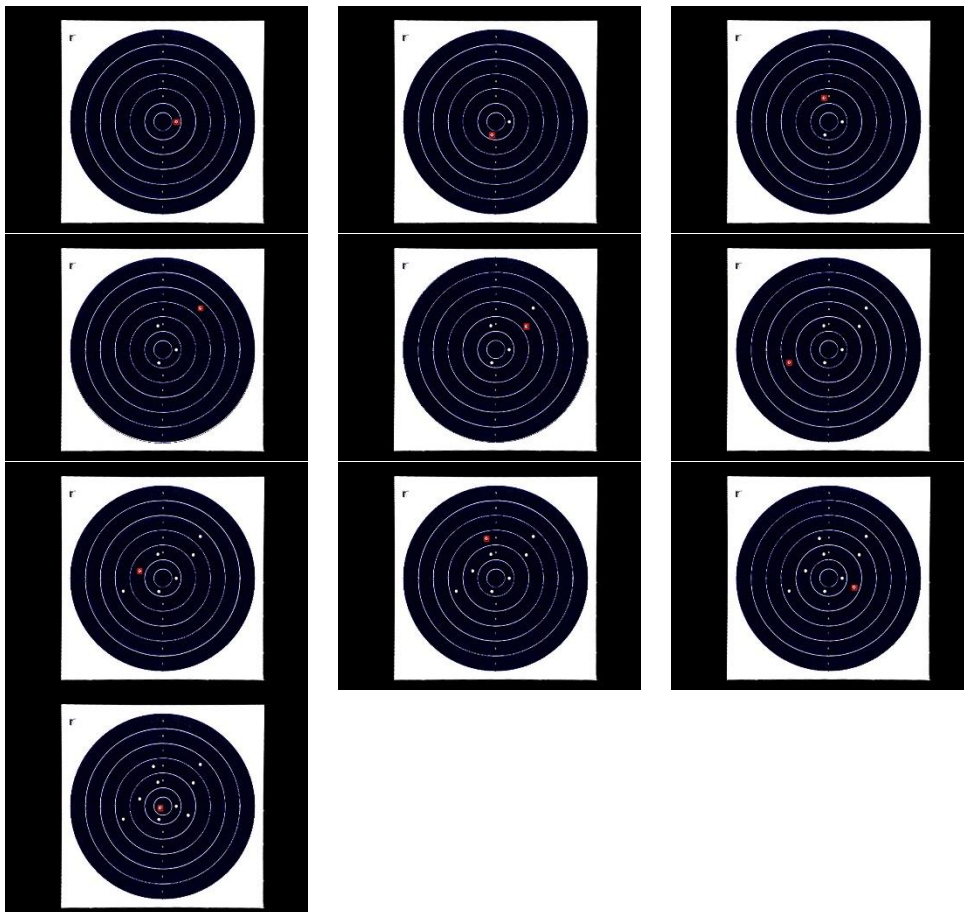


Data Ke-5

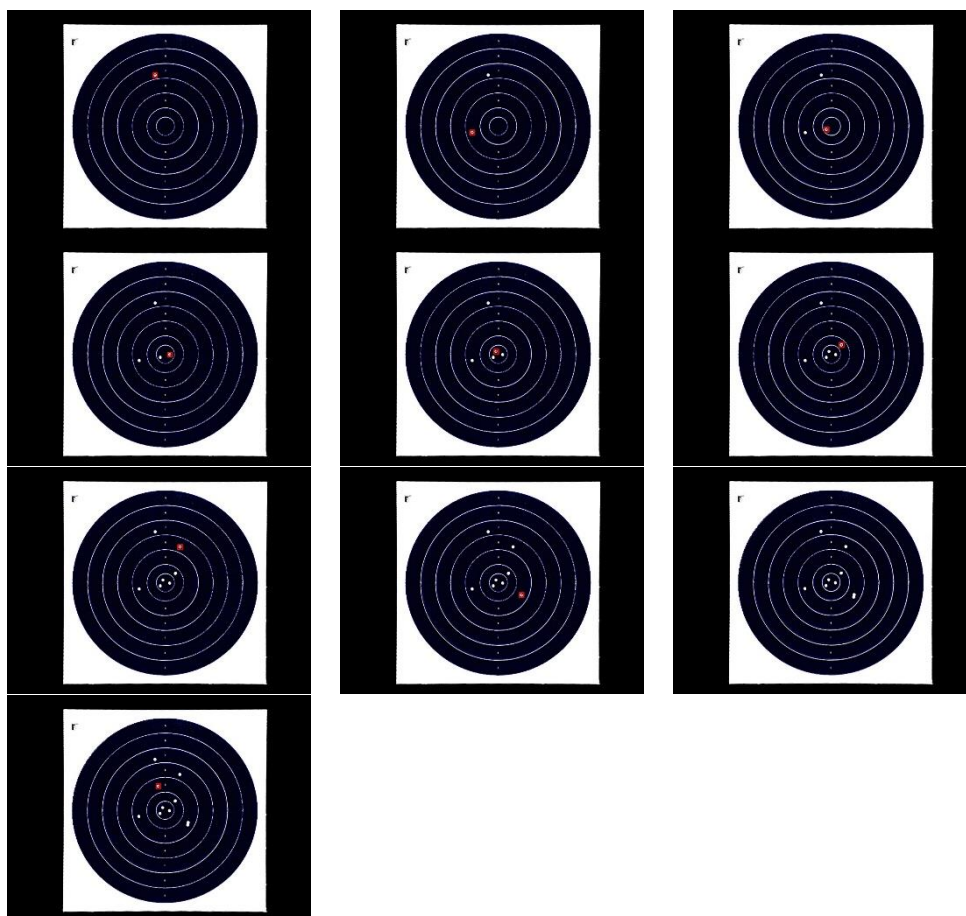




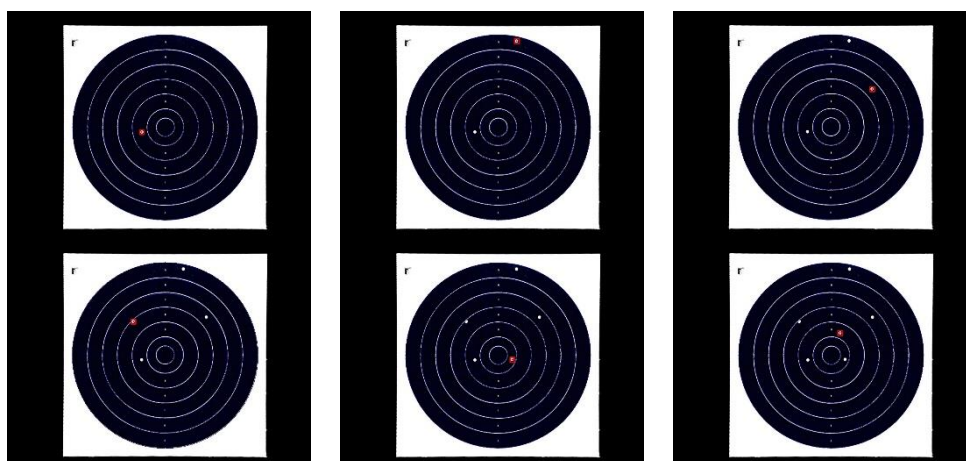
Data Ke-6

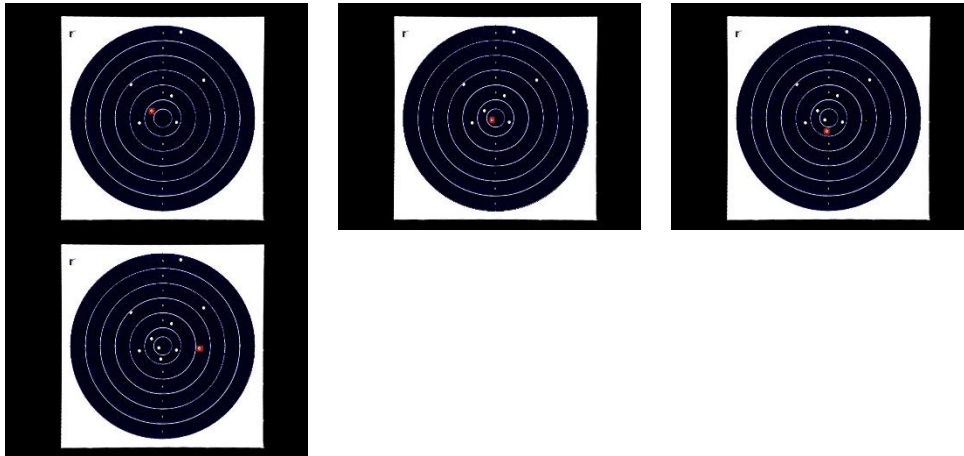


Data Ke-7

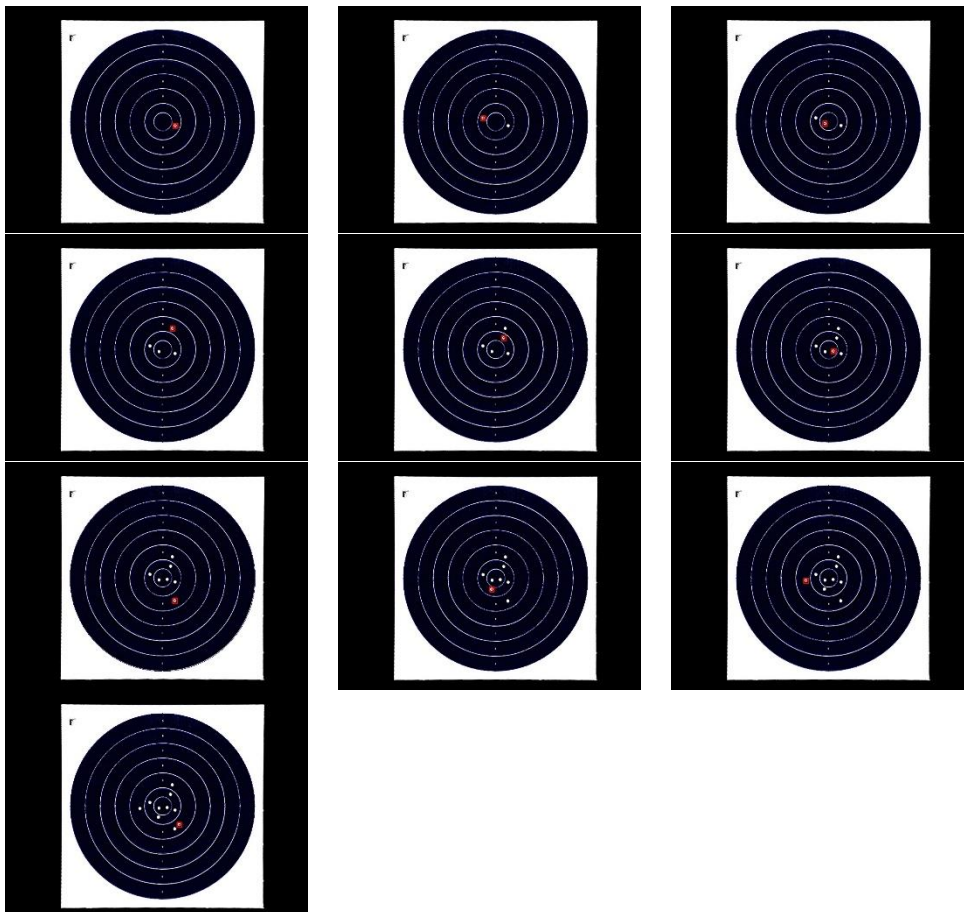


Data Ke-8

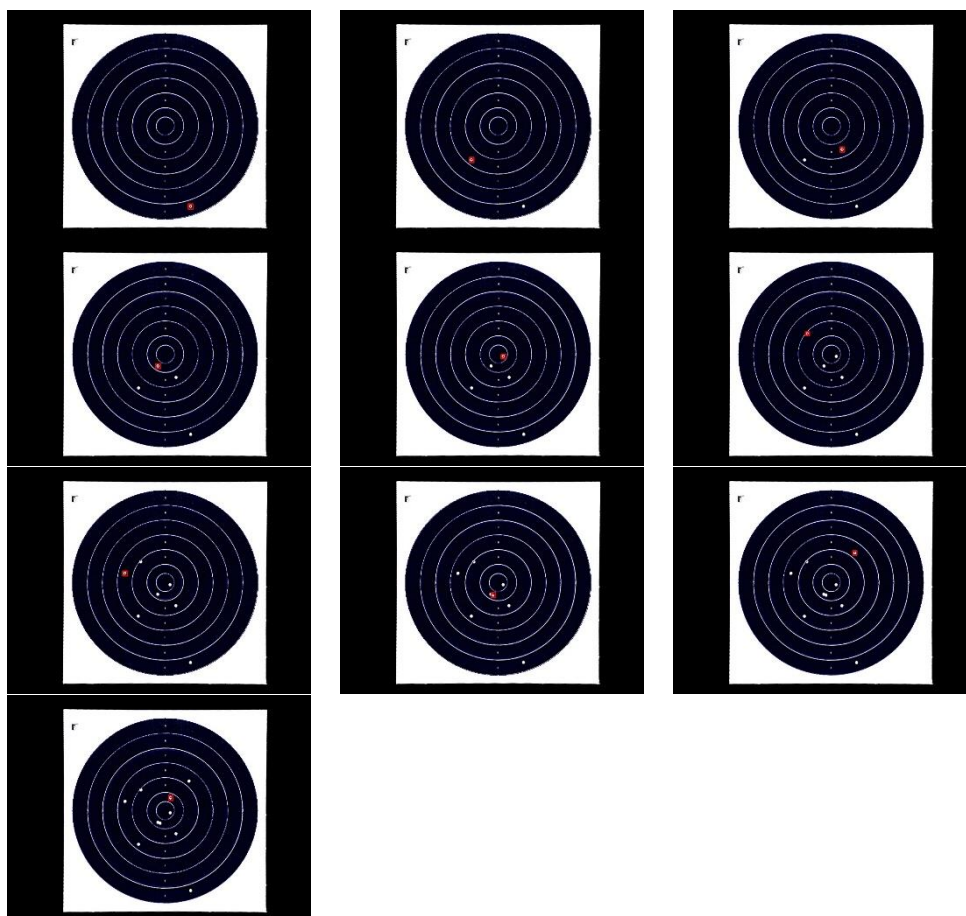




Data Ke-9



Data Ke-10



Halaman ini sengaja dikosongkan

BIOGRAFI PENULIS



Parama Diptya Widayaka dilahirkan di Pamekasan, 14 Januari 1994. Merupakan anak tunggal dari pasangan Drs. Lilik Yulianto dan Lailatul Qomariyah. Penulis memulai pendidikan di SD Negeri Bugih III Pamekasan, lalu melanjutkan di SMP Negeri I Pamekasan. Penulis menempuh jenjang pendidikan menengah atas di SMA Negeri 1 Pamekasan. Pada tahun 2012 penulis menempuh pendidikan diploma di Politeknik Elektronika Negeri Surabaya dengan bidang keahlian Teknik Mekatronika. Setelah menyelesaikan studi di tahap diploma, pada tahun 2017 penulis meneruskan pendidikan di Program Magister Jurusan Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya dengan bidang keahlian Elektronika Industri.

e-mail : diptyawidayaka@gmail.com

CP : 0812 5251 2853